LaTeX Course 2011

$\mathcal{F}$rom βasics to αdvanced τypesetting

Arho Virkki

VTT Technical Research Centre of Finland

2011-10-04

About the Course

Course Page:

http://cell.vtt.fi/latex

Suggested preliminary knowledge:

original:	http://tobi.oetiker.ch/lshort/lshort.pdf
copy:	http://cell.vtt.fi/latex/extras/lshort.pdf

Suomeksi:

alkup.:	http://www.tex.ac.uk/CTAN/info/lshort/finnish/lyhyt2e.pdf
kopio:	http://cell.vtt.fi/latex/extras/lyhyt2e.pdf

About the Course...

Suggested book:
Helmut Kopka and Patrick W. Daly, Guide to LaTeX, 4th edition.
In my opinion, the book is very thoroughly written and works perfectly as a reference.

Other suggested books:
The LaTeX Companion, The LaTeX Web Companion, The LaTeX Graphics Companion
These are the fundamental and also the most often cited books. Their only disadvantage is that there are even more pages to read through.

About the Course...

There appears not to be both a **complete** and a **concise** electronic reference to the system. In contrast, the amount of documentation is huge, and most of it is more or less outdated.

$\rightarrow$ The book is a valuable investment.

Some of the electronic references are rather complete: teTeX hypertext help is rather complete, and the TeXlipse plug-in for Eclipse knows quite the well the basic commands.

About the Course...

Most of the texts in natural sciences are written with LaTeX nowadays.

LaTeX is actually part of the scientific heritage, including the other open sourced tools like Linux, PostgreSQL, the BSD-system, Apache, etc.

Keywords: markup languages, GNU and open source.

What is LaTeX?

LaTeX, or precisely LaTeX 2_ε, is a typesetting system for structured documents, including

- books
- scientific papers
- handouts
- research reports
- slides — like now.

What is LaTeX?...

The document format is structured, which means that there are

- logical and (preferably only few)
- typographical

commands mixed with the manuscript text[1].

The text-formatted source can be compiled into different formats, the most common being

- PDF (Portable Document Format) and
- HTML (HyperText Markup Language).

[1] The manuscript has the role of “source code” in this context.

This course is for you, if you. . .

This course is for you, if you. . .

- compose books,
- publish in natural sciences,
- compose mathematical texts,
- generate (pdf) documents automatically from software,
- just like software that (just) works, or
- just like free / open source software.

Course contents

Course contents

1. short introduction
2. logical constructs: lists, environments,...
3. typographical building blocs: lines, boxes, fonts,...
4. mathematical typesetting: $\oint_C \frac{3\pi}{\sqrt[n]{\frac{z-1}{z+1}}} \mathrm{dz}$, ...
5. graphics: adding and drawing images,...
6. other tools: BibTeX for references, drawing images,...

Welcome to the Course!

$$X\beta = y + \epsilon \quad \Leftrightarrow \quad \begin{bmatrix} 1 & x_{11} & \cdots & x_{1M} \\ 1 & x_{21} & \cdots & x_{1M} \\ \vdots & & & \vdots \\ & & & \\ \vdots & & & \vdots \\ 1 & x_{N1} & \cdots & x_{NM} \end{bmatrix} \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_M \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ \\ \vdots \\ y_N \end{bmatrix} + \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \\ \vdots \\ \epsilon_N \end{bmatrix}$$

LaTeX Course 2011

Part 1: Short Introduction

Arho Virkki

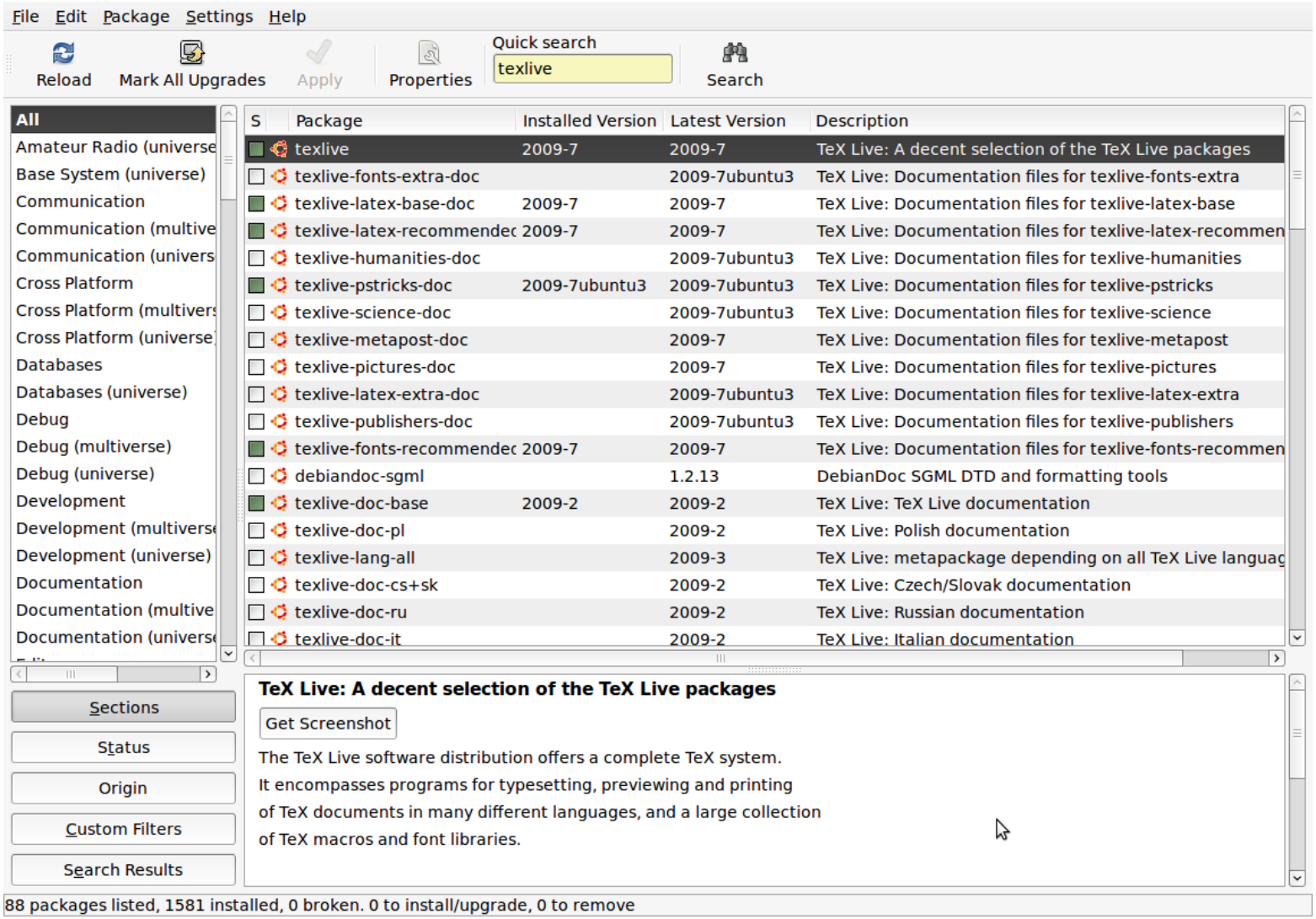
VTT Technical Research Centre of Finland

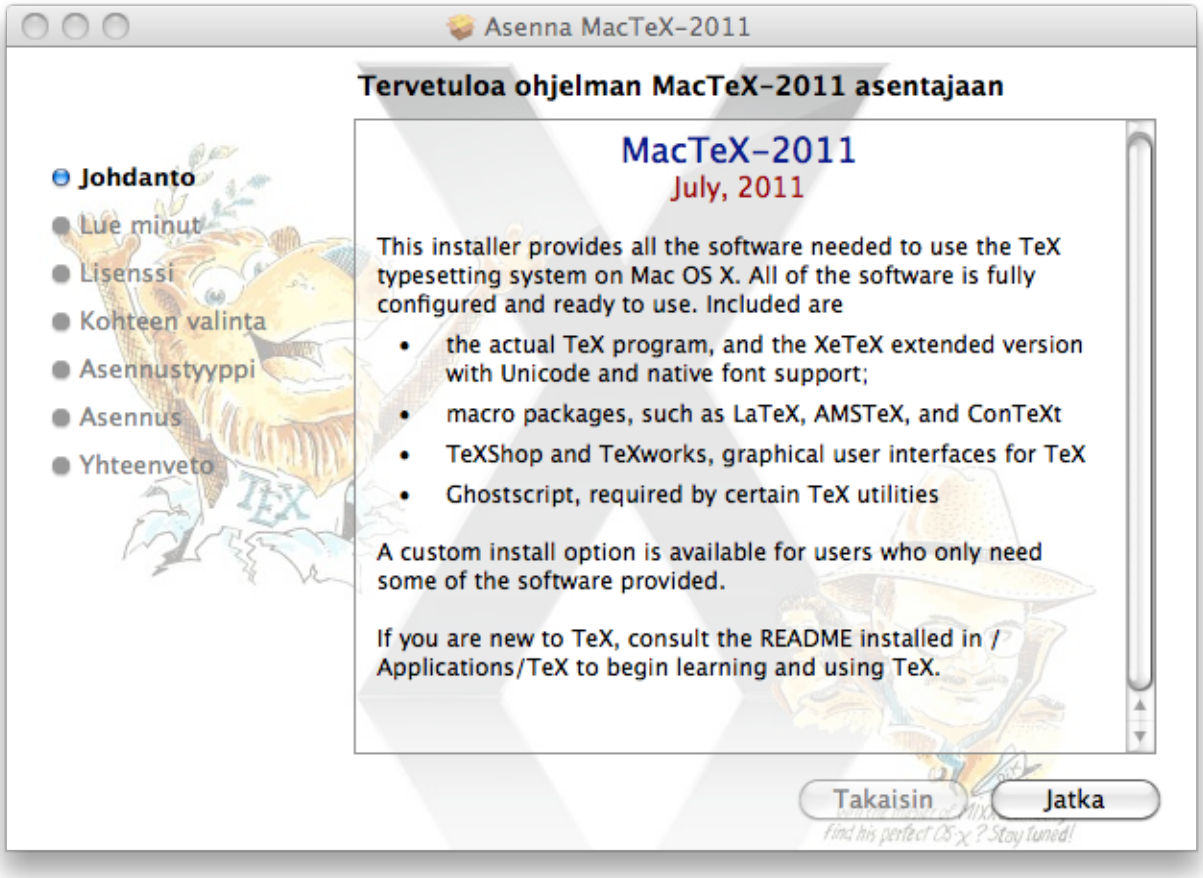
LaTeX–distribution packages

The very same LaTeX in different packages:

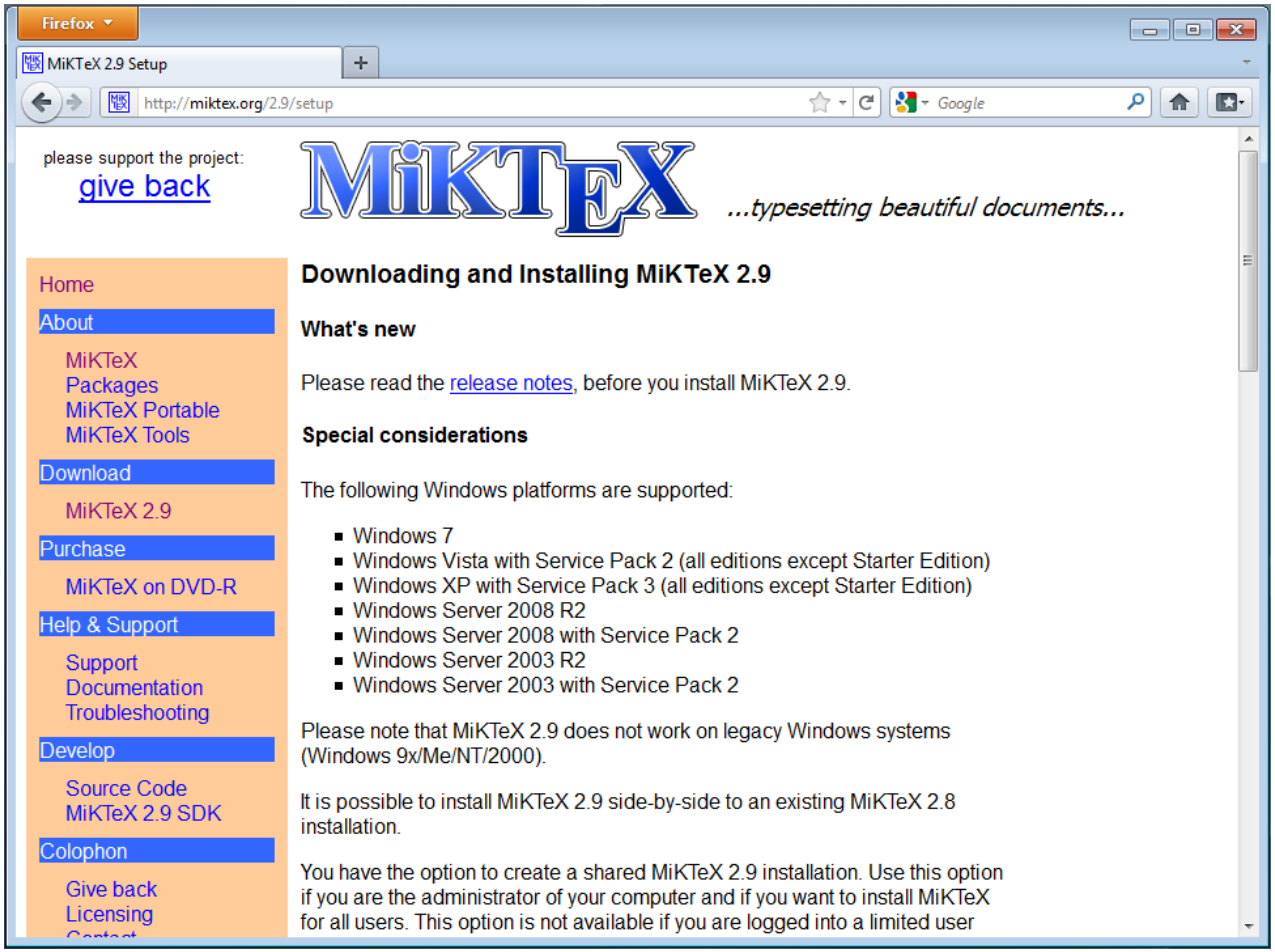
- TeXLive (Linux, Mac OS X, Windows)
- MacTeX (Mac OS X)
- MikTeX (Windows)
- ...

LaTeX on Ubuntu Linux through Synaptic




Asenna MacTeX-2011
Tervetuloa ohjelman MacTeX-2011 asentajaan
Johdanto
Lue minut
Lisenssi
Kohteen valinta
Asennustyyppi
Asennus
Yhteenveto
MacTeX-2011
July, 2011
This installer provides all the software needed to use the TeX typesetting system on Mac OS X. All of the software is fully configured and ready to use. Included are
the actual TeX program, and the XeTeX extended version with Unicode and native font support;
macro packages, such as LaTeX, AMSTeX, and ConTeXt
TeXShop and TeXworks, graphical user interfaces for TeX
Ghostscript, required by certain TeX utilities
A custom install option is available for users who only need some of the software provided.
If you are new to TeX, consult the README installed in / Applications/TeX to begin learning and using TeX.
Takaisin
Jatka

LaTeX on Windows



http://miktex.org/2.9/setup

Document structure

An example document preamble:

```
\documentclass[a4paper,10pt]{article} % style
\usepackage[latin1]{inputenc} % or [ansinew]
\usepackage[finnish]{babel}   % hyphenation
\usepackage{graphicx}         % for images

\begin{document}
\section{Introduction}
During the last few decades, the amount of digital
content has increased enormously
\dots
\end{document}
```

Compilation: Linux & Mac command line

Just call LaTeX or PDFLaTeX directly

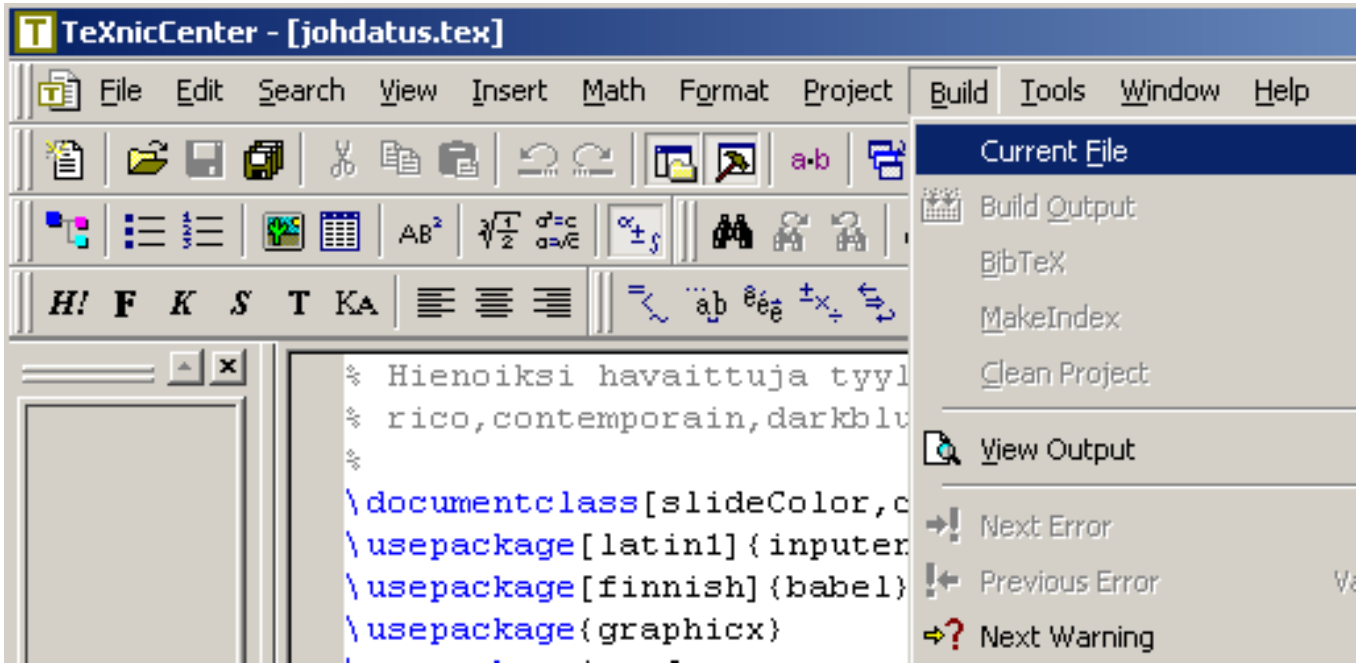
```
$ latex myfile.tex; dvipdf myfile  # or
$ pdflatex myfile
```

and then inspect the content:

```
$ gnome-open myfile.pdf # Linux
$ open myfile.pdf       # Mac
```

Compilation: Windows

Point and click - depending on the chosen interface



Mathematics

Equations are written between the $ -characters. For example,

`$\sqrt{x^3}$` $\mapsto$ $\sqrt{x^3}$

or separately as

```
\[  \sqrt{x^3}  \]
```

or using

```
\begin{equation}
\sqrt{x^3}
\end{equation}
```

where the latter gives the equation also a number.

Mathematics...

Examples:

```
\begin{equation}\label{eq:gammaf}
  \Gamma (n) :=
  \int_0^\infty x^{n-1}e^{-x} dx
\end{equation}
Observe that (\ref{eq:gammaf}) does not converge
when $n=0$
```

$$\Gamma(n) := \int_0^\infty x^{n-1}e^{-x}dx \tag{1}$$

Observe that (1) does not converge when $n = 0$.

Mathematics...

```
\[ \neg A := X \setminus A \]
```

$$\neg A := X \setminus A$$

```
\[ \zeta(s) :=
   \sum_{k=1}^\infty \frac{1}{k^s} \]
```

$$\zeta(s) := \sum_{k=1}^{\infty} \frac{1}{k^s}$$

Mathematics...

It is best to learn the basic commands

```
\frac{}{}
\int
\sum
\dots
```

by heart. There are just a few of them and they are rather logical.

(Observe that `\int` = integral, not an integer...)

Structures

Structures define how the text is displayed.

Examples:

```
\begin{enumerate}
\item Firstly,
\item Secondly\dots
\end{enumerate}
```

1. Firstly,
2. Secondly...

Structures...

```
\begin{itemize}
\item gloves
\item shoes
\end{itemize}
```

- gloves
- shoes

Structures...

```
My maths teaches was a genius to explain things:
\begin{quote}
We define the determinant like a civil service
department would, in a rather boring way: It is just
thrown to your face with the motivation being
"learn it or die".
\end{quote}
```

My maths teaches was a genius to explain things:

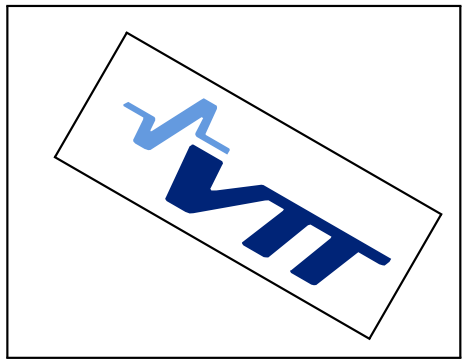
> *We define the determinant like a civil service department would, in a rather boring way: It is just thrown to your face with the motivation being "learn it or die".*

Graphics

In LaTeX, everything can be thought of being composed of boxes, aligned with respect to each other, and the distances between them.

Examples:

```
\begin{center}
\fbox{
\rotatebox{-30}{
\fbox{
\includegraphics[width=2cm]
{/img/vttplain}}}}
\vspace*{1cm}
```



Graphics...

```
\reflectbox{
\rotatebox{30}{
\resizebox{!}{5mm}{Tricky stuff}
}}

\vspace*{1cm}
\rule{3cm}{1ex}
```

Summary

- The LaTeX manuscript file consists of plain text. It takes only a little amount of disk space and it is simple to send to others.
- The manuscript always begins with the "`\documentclass[<params>]{<class>}`" command[1].
- The text consists of the commands (called tags in HTML) and the actual text.
- This is not harder that writing HTML5 / CSS3 by hand. To be honest, LaTeX is much easier!

[1] I suggest to copy the preamble from an existing template. Try to avoid the ancient templates from the early 80's, as things have changes since that.

LaTeX Course 2011

Part 2: Logical commands

Arho Virkki

VTT Technical Research Centre of Finland

Logical commands

The document is described in abstract level. Logical command define, for example,

- sections (`\section{foo}`),
- environments (`\begin{frame}`),
- emphasis (`\emph{observe}`),
- ...

and the interpretation of these commands is controlled by the style sheet (.sty) and it depends, of course, also on the media (.pdf document, .pdf slides or .html page).

Observe that LaTeX was first introduces in 1985 as an extension for TeX (first released in 1978), and HTML5/CSS3, copying the same idea, is called the "future of the web" (as of 2011).

What this means?

Setting the style by hand is not a good idea[1].

Avoid:

```
\textbf{Backround}\\[3mm]
In the future, one of the most striking
challenges will be...
```

Instead, use:

```
\section{Backround}
In the future, one of the most striking
challenges will be...
```

[1] Compare this with doing . . . in an html document instead of using <emph> and defining that environment properly in the cascading style sheet (css)!

What this also means?

- LaTeX is a sophisticated, and thus also a bit involved environment.
- You should be writing scratch files in plain text (or with paper and pencil – and then feed the papers into scanner to archive them).
- Nevertheless, I use LaTeX to write down the potential ideas still being immature for publication just to make them clean.

Headings

The levels of headings are

-1	`\part{}`
0	`\chapter{}`
1	`\section{}`
2	`\subsection{}`
3	`\subsubsection{}`
4	`\paragraph{}`
5	`\subparagraph{}`

Headings...

- The levels `\part{}` and `\chapter{}` are not used with all document types. These command are used mostly in books.
- Ending command (analog to ending tag) is not needed[2].
- Command `\setcounter{secnumdepth}{n}` $n \in 0, \ldots, 5$ defines the extent of explicit numbering (in front of the headings).

[2] This is in contrast to XML, where the elements always consist of the starting and ending tag and the content between them. For example, `<tag>` must be ended with `<\tag>`.

Environments

The most common environments are `equation`, `displaymath` and `itemize`. Suppose that `end1` and `env2` are two environments. Then

```
\begin{env1}
   ...
   \begin{env2}
   ...
   \end{env2}
\end{env1}
```

is legal. Observe that the environments need to be nested: `env1` can not end before `env2`.

Environments: Lists

```
\begin{enumerate}
\item one
\item two
\end{enumerate}
\begin{itemize}
\item[+] pros
\item[--] cons
\end{itemize}
```

yields

1. one
2. two

+ pros
– cons

Environments: Plain text

```
\begin{verbatim}
  o o
   *
  \_/
\end{verbatim}
```

produces

```
  o o
   *
  \_/
```

Environments: Plain text...

Source code listings are convenient to include with `\verbatiminput{myprog.f90}`.

For algorithm, one should use an environment that can emphasize the keywords.[3]

[3] This is a good example of an exercise for this course: Find a package for this and describe how it is used.

Environments: Text alignment

`center`:

$$\sum$$

`flushright`:

$$\sum$$

`flushleft`:

$$\sum$$

Environments: Text alignment...

```
\begin{center}
KYNTTILÄT SYTTYVÄT VARHAIN\bigskip

Kiertävät unettavat auringonnousut\\
Kaikkialle harsona niin\\
Aattoni vähiin käynyt\\
Kun kevät uutena nousee\\
\dots
\end{center}
\begin{flushright}
\emph{-- Kuusumun profeetta}
\end{flushright}
```

Environments: Text alignment...

KYNTTILÄT SYTTYVÄT VARHAIN

Kiertävät unettavat auringonnousut
Kaikkialle harsona niin
Aattoni vähiin käynyt
Kun kevät uutena nousee
...

– Kuusumun profeetta

Environments: Text alignment...

The 'quote' environment is rather basic – but it works.

```
\begin{quote}
He has a profound respect for old age.
Especially when it's bottled.
--- Gene Fowler
\end{quote}
```

> *He has a profound respect for old age. Especially when it's bottled. — Gene Fowler*

The command `quotation` works better for longer quotations.

Environments: Table

```
\begin{tabular}{|l|c|c|}
\hline
Name & J. Foo & G. Bar \\
\hline
\hline
A-score & 1 & 3 \\
B-score & -2 & 0 \\
\hline
```

Environments: Table...

Name	J. Foo	G. Bar
A-score	1	3
B-score	-2	0

Somehow this looks shomehow crowded... Let's change the title into

```
\rule[-4pt]{0pt}{16pt}Nimi & J. Foo & G. Bar \\
```

Environments: tabular...

Name	J. Foo	G. Bar
A-score	1	3
B-score	-2	0

We extended the borders of the table with a line of with zero.

The LaTeX tables are, to be honest, rather clumsy. This is not a problem if we can generate them automatically (e.g. from the R language using the xtable package).

Environments: Custom definitions

Let us define an environment than adds automatically a box at the end of a proof, and writes the title 'New Proof.' in the beginning:

```
\newenvironment{newproof}
{\makebox[2cm][l]{\textbf{New Proof.\ }}}
{\hfill $\Box$}
```

Environments...

```
\begin{newproof}
Suppose that $\epsilon > 0$ is already
chosen,\dots
\end{newproof}
```

New Proof. Suppose that $\epsilon > 0$ is already chosen, ... □

Environments...

The general form of the environment definition:

```
\newenvironment{name}[args]{begdef}{enddef}
```

or

```
\renewenvironment{name}[args]{begdef}{enddef}
```

if we wish to re-define an existing environment.

Environments...

These exists a command for stating lemmas:

```
\newtheorem{command name}{theorem name}[counter]
```

For example,

```
\newtheorem{newlemma}{Lemma}[page]

\begin{newlemma}[The Professors' Remainder
Theorem]
If we use half of the time left for each
exercise, we can deal with an infine number
of exercises.
\end{newlemma}
```

Environments...

Lemma (The Professors' Remainder Theorem)

If we use half of the time left for each exercise, we can deal with an infine number of exercises.

Let us further inspect the command:

```
\newtheorem{newlemma}{Lemma}[page]
```

where the numbering was set to follow the `page` counter. Thus, (if this was an article and not a slideshow) all formulas are numbered as

(page,lemma)

Environments...

```
\newtheorem{mynote}{Note!}
\begin{mynote}
Huom\dots
\end{mynote}
```

Note!

Notes lose their meaning if there are too many of them, or the notes are self-evident.

Environments...

Note!

LaTeX has lots of counters that can be cited with the command `\the<counter_name>`*:*

```
We are on slide \thepage.
```

We are on slide 24.

Environments: Summary

Remember the commands

- `\newcommand` and
- `\newtheorem`.

Details can be found from books or on-line help systems.

You can inspect e.g. http://en.wikibooks.org/wiki/LaTeX,
or by typing the command name into Google — which seems to be a reasonable default procedure for anything nowadays.

Commands

If we need a simple command, e.g. `\warning{text}`, one can define

```
\newcommand{\warning}[1]{%
\begin{center}
\Large
\shadowbox{
\textbf{#1}}
\end{center}
}
```

Commands...

Now, the command

```
\warning{orthogonal vectors $\neq$
orthonormal vectors}
```

produces

orthogonal vectors $\neq$ orthonormal vectors

Observations

- `\shadowbox{}` is from the package `fancybox`. There are tons of similar useful and less useful packages available at the Comprehensive TeX Archive Network (CTAN).
- To define new high level logical commands, one of course needs to know something about the low level LaTeX "programming".

LaTeX Course 2011

Part 3: Typographical commands

Arho Virkki

VTT Technical Research Centre of Finland

Elements

The non-textual elements, including

- boxes (`\makebox, \framebox ...`)
- lines (`\rule `)
- distances (`\hspace, \hspace*, \vspace, ...`)
- ...

are described with commands inside the document.

Their arguments are given in millimeters, centimeters, pixels, or in proportion of the font 'M' or 'x' characters,

10mm, 1.5cm, 12px, 1em, 0.8ex, ...

Examples

Let us inspect some of the most common commands:

- `\rule[voffset]{width}{height}`
  e.g.: `Baseline text\rule[-5pt]{2cm}{1ex}`:
  Baseline text  .

- `\framebox[width][pos]{text}`
  e.g.: `\framebox[9cm][s]{foo  bar baz}`:

  foo bar baz

  where s=stretch.The other options are l=left and r=right

Examples...

- `\makebox[width][pos]{text}`
  is the same as `framebox`, but with no borders
- `\fbox{text}`
  is a short form: `\fbox{Inspect this!}`
  Inspect this!
- `\mbox{text}`
  The invisible box `\mbox{the contents}`
  $\mapsto$ the contents is treated as a single entity. This property is useful, and will be needed many times in the future...

Examples...

- `\vspace*{height}` ja `\hspace*{width}` `\hspace*{0.3\textwidth}` adds certain amount of space to the documents[1][2]. Vertical space can be added with `\vspace`. (Negative distances move the following contents to the left or up. The command with star is enforcing, otherwise adding the space is optional, and will be omitted at certain places, e.g. at the end of paragraph)
- `\parbox[pos][height][ipos]{width}{text}` produces a box. For example,

```
The \fbox{\parbox[b][1cm][b]{4cm}{Schrödinger's cat}}
```

The Schrödinger's cat

[1] Observe the relative distance measure `\textwidth`

[2] Footnotes can be produces with `\footnote...`

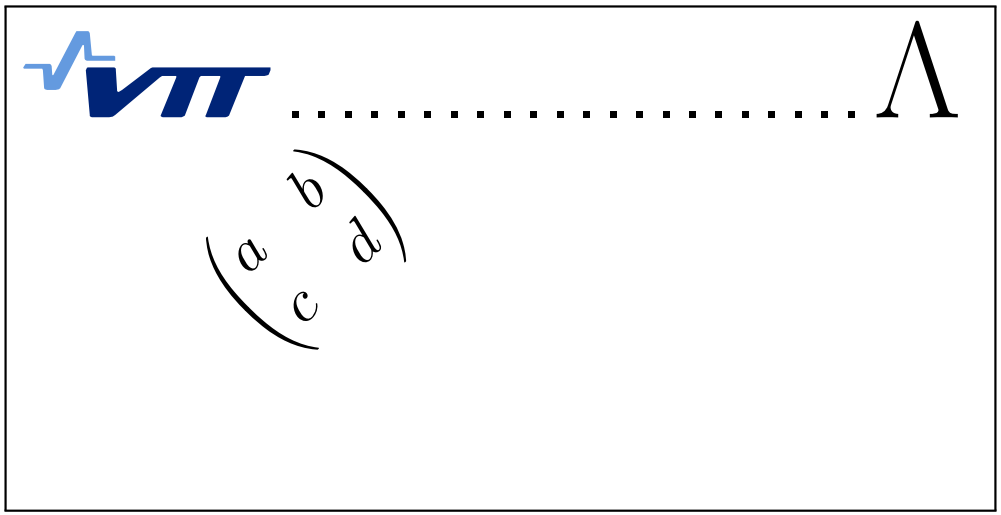
Examples...

- `pos` can be t=top (line) or b=bottom (line)
- `ipos` can be t=top, b=bottom, c=centered or s=stretched
- `parbox` behaves like a small embedded page[3], end it is treated as a single element.

[3] `\minipage` will behave exactly like an embedded page...

Examples...

```
\fbox{\parbox[t][3cm][t]{6cm}{%
    \includegraphics[width=1.6cm]
    {img/vttplain} \dotfill {\Huge $\Lambda$} \\
    \hspace*{1cm}\rotatebox{45}{%
      $\scriptstyle\begin{pmatrix}
        a & b \\ c & d \end{pmatrix}$}}
}
```



Examples...

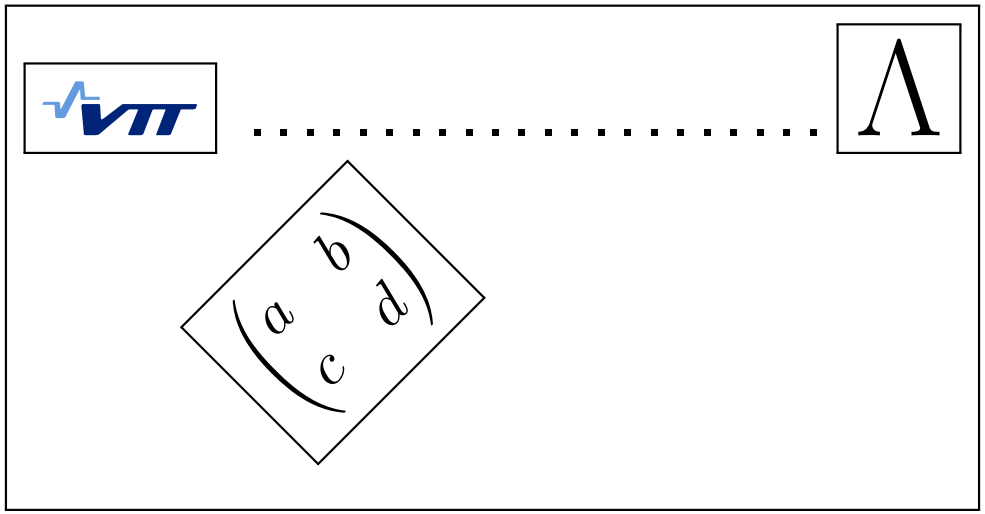
What just happened?

- `\fbox` included borders to the box (in contrast to mbox)
- `\includegraphics` included the image (as pdf or png)
- `\dotfill` fills the area with dots
- `\hspace*` moved the «pencil« horizontally

The following shows the same example with all elements having a box around them.

Examples...

Now we see what happens



The only difference is that the borders take some space, which makes a small difference in the image.

Conclusions

We saw that LaTeX code becomes complicated, if we construct detailed elements.
Because of this reason, the logical content of the document and fiddling with the details should be kept **separated**.
In summary,

- everything can be tuned,
- it is not wise to tune everything,

but from experience I suppose that you would like to change the default font sooner or later. . .

Fonts

The easiest way to get started with different document styles is to load a pre-defined style package (from a `.sty` file) in the manuscript preamble:

```
\usepackage{mystyle}
```

where the style is possibly one from the list `helvet`, `palatino`, `avant`, `charter`, `bookman`, `newcent` or `times` (that are probably already installed in your LaTeX-distribution package).

The size of the font can be changed with the descriptive commands `{\large }` and `{\small }`.

Fonts...

`{\tiny Rabbit}`	Rabbit
`{\scriptsize Rabbit}`	Rabbit
`{\footnotesize Rabbit}`	Rabbit
`{\small Rabbit}`	Rabbit
`{\normalsize Rabbit}`	Rabbit
`{\large Rabbit}`	Rabbit
`{\Large Rabbit}`	Rabbit
`{\LARGE Rabbit}`	Rabbit
`{\huge Rabbit}`	Rabbit
`{\Huge Rabbit}`	Rabbit

Fonts...

Observations

- These commands are declarations, not functions: `\large` = change to large font and keep in until the end of this (scope or) environment.
- Anything between the `\begin{}` ... `\end{}` -block is an environment. A new environment can also be defined with the curly braces { ... }.
- Use the curly braces and think of the environments. This is better style (from the programmer's point of view) than scattering the document with commands like `\normalsize` without a clear reference to the beginning of the "non-normal" behaviour.

Fonts...

I admit that even the last one `\Huge` was not *that* large:

Q: How to do a HUGE Rabbit?

A: `\scalebox{4}{Rabbit}`

Fonts...

In addition, there are the ordinary functions to decorate the text

`\texttt{Rabbit}`	`Rabbit`
`\textit{Rabbit}`	*Rabbit*
`\textbf{Rabbit}`	**Rabbit**
`\textsl{Rabbit}`	*Rabbit*
`\textsc{Rabbit}`	RABBIT
`\underline{Rabbit}`	Rabbit

Fonts...

These commands suffice well for a sophisticated writer.

More fonts can be found from the LaTeX-distributions by searching the `.fd` (font definition) files. For example, `t1pbk.fd` (T1 encoded postscript Bookman font definition) can be selected with the definition

```
\fontfamily{pbk}\selectfont
This is Bookman Font
```

This is bookman Font

Fonts...

If you really need to change the font all the time (which is, in general, a bad idea) , it may be advantageous to define a custom command in the beginning of the document :

```
\newcommand{\avantgar}[1]{
{\fontfamily{pag}\selectfont #1}}
```

etc...

```
\newcommand{\bookman}[1]{{\fontfamily{pbk}\selectfont #1}}
\newcommand{\courier}[1]{{\fontfamily{pcr}\selectfont #1}}
\newcommand{\cmodern}[1]{{\fontfamily{cmr}\selectfont #1}}
\newcommand{\helvetic}[1]{{\fontfamily{phv}\selectfont #1}}
\newcommand{\newcent}[1]{{\fontfamily{pnc}\selectfont #1}}
\newcommand{\tmroman}[1]{{\fontfamily{ptm}\selectfont #1}}
\newcommand{\script}[1]{{\fontfamily{pzc}\selectfont #1}}
```

Fonts...

Now we have defined a couple of commands that fit the LaTeX philosophy.

`\avantgar{Rabbit}`	Rabbit
`\bookman{Rabbit}`	Rabbit
`\courier{Rabbit}`	Rabbit
`\cmodern{Rabbit}`	Rabbit
`\helvetic{Rabbit}`	Rabbit
`\tmroman{Rabbit}`	Rabbit
`\script{Rabbit}`	*Rabbit*

These commands are not included by default, because they will most likely ruin your (technical or scientific) document. For the purposes you would be using them, some of the office packages might be a better option.

Conclusions...

A good strategy for writing a paper with LaTeX is to

- Make the title page by hand (e.g. by using the command just learned)
- Write the text with clean LaTeX, using perhaps some of the extension packages from AMS (American Mathematical Society) to stick with portability.
- Make your own style sheet, if necessary
- Include minor things in the document preamble before the `\begin{document}` clause (even though it is sometimes legal to define command later in the document — but what is the advantage?)

Conclusions...

- LaTeX and plain TeX spaghetti mixed with the text is a nightmare to maintain. Recall the experiences from HTML 3.2 before properly working CSS.
- LaTeX is a large software with a very long history (starting from 1978). Knowing the basics suffices well.

LaTeX Course 2011

Part 4: Typesetting mathematics

Arho Virkki

VTT Technical Research Centre of Finland

Typesetting mathematics

Inlined equations are written inside `$$` characters:
E.g., `$\sum_{k=1}^\infty a_k$` produces $\sum_{k=1}^\infty a_k$.

Equations on their own line are put into a separate environment `\begin{equation}..\end{equation}`[1],

$$\sum_{k=1}^\infty a_k, \tag{1}$$

or `\[..\]`[2],

$$\sum_{k=1}^\infty a_k.$$

[1] This produces a number.
[2] This doesn't.

Typesetting mathematics...

Observe the differences in style when writing the equations within the text, or separately (e.g. the limits of the sum were written differently)

The style can also be forced with the commands,

- `\textstyle`
- `\displaystyle`

if needed.

Typesetting mathematics...

In the math mode, letters (the variables or constants) are typeset with slanted typeface and extra spaces are ignored:

```
\begin{equation}
I(x) := a_1^2 b_{1,2} c x .
\end{equation}
```

$$I(x) := a_1^2 b_{1,2} c x. \tag{2}$$

If the equations are part ot the sentence, commas should be written such that the formulas become a natural part of the sentence.

Typesetting mathematics...

The characters

```
# $ % & ~ _ ^ \ { }
```

have a special meaning in LaTeX. If they are needed in the equations, one must write

```
\#   \$   \%   \&   \_   \{   \}
```

For example,

```
\[ \mathcal{L}\{f\}(s) :=
   \int_0^\infty f(t) e^{-st} dt \]
```

$$\mathcal{L}\{f\}(s) := \int_0^\infty f(t)e^{-st}dt$$

Typesetting mathematics...

- Because ordinary keyboards are lacking most of the mathematical symbols, there are quite a number of special commands – in addition to the greek letters
  $\alpha, \beta, \gamma, \delta,\ \epsilon, \zeta, \eta, \theta, \iota,\ \kappa, \lambda, \mu, \nu, \xi, o, \pi, \rho, \sigma, \tau, \upsilon, \phi, \chi, \psi$ and ω
- The commands for mathematical typesetting are best learned by heart. Editors (including Eclipse) also help in remembering the commands.

Upper and lower indices

The upper and lower indices can be produced with ^ and _.

```
\[ V^2_1 \ge V_{21} \]
```

$$V^2_1 \ge V_{21}$$

If there are more than one index, they need to be grouped with curly braces.

Rational expressions

```
\[ \frac{a}{b} \]
\[ \sqrt{2} \approx 1.4142 \]
```

$$\frac{a}{b}$$

$$\sqrt{2} \approx 1.4142$$

Rational expressions...

```
\[ \left(
   \frac{\sqrt[3]{z-1}}{2}
   \right)^2 \]
```

$$\left(\frac{\sqrt[3]{z-1}}{2}\right)^2$$

Rational expressions...

Revision:

- Fraction: `\frac{numerator}{denominator}`
- Root: `\sqrt{}` ja `\sqrt[n]{}` (!)
- The size of the parenthesis can be automatically adjusted with the command pair `\left(` and `\right)`
  - Also `\left[, \left\{` and `\left|` work
  - One can mix $|, (, [, \{$ and .
  - The dot shows nothing. Example: $\left\{ \frac{\frac{a}{b}}{C} \right.$

Rational expressions...

Chain fractions (or continuous fractions):

```
\[ a_0 + \cfrac{1}
         {a_1 + \cfrac{1}
                {a_2+\frac{1}{a_3}}} \]
```

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \frac{1}{a_3}}}$$

Differential operators

```
\[ \frac{dy}{dt} = f(y;\lambda) \]
\[ \frac{\partial u}{\partial t} = \kappa^2
   \left( \frac{\partial^2 u}{\partial x^2}
        + \frac{\partial^2 u}{\partial y^2}
        + \frac{\partial^2 u}{\partial z^2}
     \right) \]
```

$$\frac{dy}{dt} = f(y;\lambda)$$

$$\frac{\partial u}{\partial t} = \kappa^2 \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right)$$

Limits, sums, integrals

```
\[ \lim_{x \to 0}
   \frac{3x^2 +7x^3}{x^2 +5x^4} = 3.\]
```

$$\lim_{x \to 0} \frac{3x^2 + 7x^3}{x^2 + 5x^4} = 3.$$

```
\[ \sum_{k=1}^\infty \frac{1}{2^k} =
   \frac{\pi^2}{6} \]
```

$$\sum_{k=1}^{\infty} \frac{1}{2^k} = \frac{\pi^2}{6}$$

Limits, sums, integrals...

```
\[ \int_{-\infty}^\infty \sin(x)\ dx = \pi \]
```

$$\int_{-\infty}^{\infty} \sin(x)\, dx = \pi$$

```
\[ \oint_{\partial C} h(z)\ dz = 0 \]
```

$$\oint_{\partial C} h(z)\, dz = 0$$

Limits, sums, integrals...

```
\[ \int\limits_{-\infty}^\infty
   \sin(x)\ dx = \pi \]
```

$$\int_{-\infty}^{\infty} \sin(x)\, dx = \pi$$

```
\[ \oint\limits_{\partial C} h(z)\ dz = 0 \]
```

$$\oint_{\partial C} h(z)\, dz = 0$$

Revision and some trivia

- The upper and lower indices are obtained with `^` ja `_`
- Spaces in equations can be enforced with `\` (backslash and space)
- The command `\limits` puts more emphasis on the upper and lower indices

Spaces in equations

```
\[ a\, b \]
\[ a\: b \]
\[ a\  b \]
\[ a\! b \]
```

$$a\, b$$

$$a\: b$$

$$a\ b$$

$$a\! b$$

- The last command `\!` is a negative space.
- Use these fine-tunings sparingly!

Math fonts

Math mode is not for writing text. Nevertheless, there are some commands to change the appearance of the fonts.

`A`	A
`\mathrm{A}`	A
`\mathbf{A}`	$\mathbf{A}$
`\mathbb{A}`	$\mathbb{A}$
`\mathcal{A}`	$\mathcal{A}$,

The last two commands only work for capital letters.

Font style, examples

```
\[ \dot{M}_{\mathrm{CO_2}} \]
```

$$\dot{M}_{\mathrm{CO_2}}$$

```
\[ \alpha \in \mathbb{C}, \mathbf{v} \in X
    \Rightarrow \alpha \mathbf{v} \in X \]
```

$$\alpha \in \mathbb{C}, \mathbf{v} \in X \Rightarrow \alpha \mathbf{v} \in X$$

- Chemical formulas are written with capital letter.
- (In modern literature, vectors are typically written just as $v \in X$, without any additional decorations.

Font size

The size of the font is also controlled differently:

`{\scriptscriptstyle \sum}`	$\scriptscriptstyle \sum$
`{\scriptstyle \sum}`	$\scriptstyle \sum$
`{\textstyle \sum}`	$\textstyle \sum$
`{\displaystyle \sum}`	$\displaystyle \sum$

Function names

LaTeXknows the most conventional function names:

```
\[ \cos (\pi) = -1 \]
```

$$\cos(\pi) = -1$$

```
\[ \sin (0) = 0 \]
```

$$\sin(0) = 0$$

Function names...

Known names include

```
\arccos \cos  \csc    \exp  \ker    \limsup
\min    \sinh \arcsin \cosh \def    \gcd
\lg     \ln   \Pr     \sup  \arctan \cot
\det    \hom  \lim    \log  \sec    \tan
\arg    \coth \dim    \inf  \liminf \max
\sin    \tanh
```

Function names...

Own functions can be **declared** as operators with the command `\operatorname`.

```
\[ \operatorname{arg min}_
   \Theta f(\Theta) \]
```

$$\operatorname{arg\,min}_{\Theta} f(\Theta)$$

Function names...

The previous example still needs some tuning:

```
\[ \operatorname{arg\ min}_
   \Theta f(\Theta) \]
```

$$\operatorname{arg\,min}_{\Theta} f(\Theta)$$

```
\[ \underset{\Theta}
   {\operatorname{arg\ min}} f(\Theta) \]
```

$$\underset{\Theta}{\operatorname{arg\,min}} f(\Theta)$$

Mixing text and formulas

LaTeX :

```
\[ S := \{ x\in \Omega | f(x) = c
          \ \mathrm{and}\ g(x) < 0 \} \]
```

$$S := \{x \in \Omega | f(x) = c, \text{ and } g(x) < 0\}$$

AMS-LaTeX:

```
\[ S := \{ x\in \Omega | f(x) = c
          \text{ and } g(x) < 0 \} \]
```

$$S := \{x \in \Omega | f(x) = c, \text{ and } g(x) < 0\}$$

Three dots(...)

Three dots:

```
\[ a_0 + a_2 + \cdots + a_n \]
\[ \ldots, \cdots, \ddots, \vdots \]
```

$$a_0 + a_2 + \cdots + a_n$$

$$\ldots, \cdots, \ddots, \vdots$$

Mathematical accents

`\underline{a}`	$\underline{a}$	`\overline{a}`	$\overline{a}$
`\hat{a}`	$\hat{a}$	`\check{a}`	$\check{a}$
`\tilde{a}`	$\tilde{a}$	`\acute{a}`	$\acute{a}$
`\grave{a}`	$\grave{a}$	`\dot{a}`	$\dot{a}$
`\ddot{a}`	$\ddot{a}$	`\breve{a}`	$\breve{a}$
`\bar{a}`	$\bar{a}$	`\vec{a}`	$\vec{a}$

Norm and absolute value

Norm:

```
\[ \|a x\| = |a| \|x\| \]
```

$$\|ax\| = |a|\|x\|$$

Parentheses

Dot produces an invisible parenthesis:

```
\[ \left( \frac{1+z}{1-z} \right. \]
\[ \left. \frac{du}{dx} \right|_{x=0} \]
```

$$\left(\frac{1+z}{1-z} \right.$$

$$\left. \frac{du}{dx} \right|_{x=0}$$

Multiline equations

```
\begin{eqnarray}
F(x) &= & \int_a^b I(u,x) du  \nonumber \\
     &= & \frac{1}{\sqrt{\pi}}G(x)
\end{eqnarray}
```

$$
\begin{aligned}
F(x) &= \int_a^b I(u,x)du \\
&= \frac{1}{\sqrt{\pi}}G(x)
\end{aligned} \tag{3}
$$

Multiline equations...

- `eqnarray` works like a table, but it has three columns fixed.
- The numbering can be suppressed with `\nonumber`
- If the numbering is not needed at all, one can use the form `\begin{eqnarray*}...\end{eqnarray*}`

Matrices

Like the table,

```
\[ \left[\begin{array}{ccc}
    1 & 0 & 2  \\
    0 & 2 & 0  \\
    0 & 0 & a \end{array}\right] \]
```

$$\left[\begin{array}{ccc}
1 & 0 & 2 \\
0 & 2 & 0 \\
0 & 0 & a
\end{array}\right]$$

Matrices...

Using the AMS-package:

```
\[ \begin{bmatrix}
  1 & 0 & 2  \\
  0 & 2 & 0  \\
  0 & 0 & a \end{bmatrix} \]
```

$$\begin{bmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 0 & 0 & a \end{bmatrix}$$

Matrices...

...or ...

```
\[ \begin{pmatrix}
  1 & 0 & 2  \\
  0 & 2 & 0  \\
  0 & 0 & a \end{pmatrix} \]
```

$$\begin{pmatrix} 1 & 0 & 2 \\ 0 & 2 & 0 \\ 0 & 0 & a \end{pmatrix}$$

Functions defined in parts

Like with tables,

```
\[ |x| = \left\{ \begin{array}{rl}
    x & \mbox{if $x \geq 0$};\\
   -x & \mbox{if $x < 0$}.\end{array}
   \right. \]
```

$$|x| = \left\{ \begin{array}{rl} x & \text{if } x \geq 0; \\ -x & \text{if } x < 0. \end{array} \right.$$

Functions defined in parts...

Or using the AMS-package:

```
\[ |x| = \left\{ \begin{aligned}
    x & \text{ if } x \geq 0;\\
   -x & \text{ if } x < 0.\end{aligned}
   \right. \]
```

$$|x| = \begin{cases} x \text{ if } x \geq 0; \\ -x \text{ if } x < 0. \end{cases}$$

Grouping with curly braces

Equations can be annotated to help the reader:

```
\[ \underbrace{A}_{\star} \]
```

$$\underbrace{A}_{\star}$$

```
\[ \overbrace{B}^{\dag} \]
```

$$\overbrace{B}^{\dagger}$$

Grouping with curly braces...

Example:

```
\[ y' = \overbrace{
   \underbrace{Ay}_{\text{linear part}} +
   \underbrace{By^R}_{\text{nonlinear part}}
   }^{\text{the driving force}} \]
```

$$y' = \overbrace{\underbrace{Ay}_{\text{linear part}} + \underbrace{By^R}_{\text{nonlinear part}}}^{\text{the driving force}}$$

Boxes

Adding a box around an equation

```
\[ \boxed{\frac{1}{1+x}} \]
```

$$\boxed{\frac{1}{1+x}}$$

- may simplify the presentation
- . . . or make it even more complicated

Boxes...

Example:

```
\begin{eqnarray*}
\| x - z \| &= &\| x\ \boxed{-y+y} - z \|\\
          &\leq &\| x - y \| + \| y - z \|
\end{eqnarray*}
```

$$\begin{aligned} \|x - z\| &= \|x \boxed{-y + y} - z\| \\ &\leq \|x - y\| + \|y - z\| \end{aligned}$$

The AMS package

In this course, we use the commands from the AMS package without separate notification (as they have become standard over the years). The following commands are especially from the AMS package `amsmath`:

`\iint`	$\iint$
`\iiint`	$\iiint$
`\iiiint`	$\iiiint$
`\idotsint`	$\int\cdots\int$

The AMS package...

```
\[ \sum_{\substack{k=1\dots n\\
                   l=1\dots m\\
                   k\neq l}}
    a_{k,l} \]
```

$$\sum_{\substack{k=1\dots n\\ l=1\dots m\\ k\neq l}} a_{k,l}$$

The AMS package...

```
\[ \overset{*}{A}, \underset{*}{B} \]
```

$$\overset{*}{A}, \underset{*}{B}$$

```
\[ \sideset{_a^b}{_c^d}\prod \]
```

$$\sideset{_a^b}{_c^d}\prod$$

Greek letters

\alpha	α	\beta	β	\gamma	γ
\delta	δ	\epsilon	ϵ	\varepsilon	ε
\zeta	ζ	\eta	η	\theta	θ
\vartheta	ϑ	\iota	ι	\kappa	κ
\lambda	λ	\mu	μ	\nu	ν
\xi	ξ	\pi	π	\varpi	ϖ
\rho	ρ	\varrho	ϱ	\sigma	σ
\varsigma	ς	\tau	τ	\upsilon	υ
\phi	ϕ	\varphi	φ	\chi	χ
\psi	ψ	\omega	ω		

Greek capital letters

Only those letters are having a separate command, which do not have a key in the ordinary keyboard.

`\Gamma`	Γ	`\Delta`	Δ	`\Theta`	Θ
`\Lambda`	Λ	`\Xi`	Ξ	`\Pi`	Π
`\Sigma`	Σ	`\Upsilon`	Υ	`\Phi`	Φ
`\Psi`	Ψ	`\Omega`	Ω		

Symbols: relations

`\approx`	$\approx$	`\asymp`	$\asymp$	`\bowtie`	$\bowtie$
`\cong`	$\cong$	`\dashv`	$\dashv$	`\doteq`	$\doteq$
`\equiv`	$\equiv$	`\frown`	$\frown$	`\ge` or `\geq`	$\geq$
`\gg`	$\gg$	`\in`	$\in$	`\le` or `\leq`	$\leq$
`\ll`	$\ll$	`\mid` or `|`	$\mid$	`\models`	$\models$
`\neq`	$\neq$	`\ni`	$\ni$	`\notin`	$\notin$

Symbols: relations...

`\parallel`	$\parallel$	`\prec`	$\prec$	`\preceq`	$\preceq$
`\perp`	$\perp$	`\propto`	$\propto$	`\sim`	$\sim$
`\simeq`	$\simeq$	`\smile`	$\smile$	`\sqsubseteq`	$\sqsubseteq$
`\sqsupseteq`	$\sqsupseteq$	`\subset`	$\subset$	`\subseteq`	$\subseteq$
`\succ`	$\succ$	`\succeq`	$\succeq$	`\supset`	$\supset$
`\supseteq`	$\supseteq$	`\vdash`	$\vdash$		

Symbols: binary operators

`\amalg`	$\amalg$	`\ast`	$\ast$
`\bullet`	$\bullet$	`\bigcirc`	$\bigcirc$
`\bigtriangledown`	$\bigtriangledown$	`\bigtriangleup`	$\bigtriangleup$
`\cap`	$\cap$	`\cdot`	$\cdot$
`\circ`	$\circ$	`\cup`	$\cup$
`\dagger`	$\dagger$	`\ddagger`	$\ddagger$
`\diamond`	$\diamond$	`\div`	$\div$
`\mp`	$\mp$	`\odot`	$\odot$

Symbols: binary operators...

`\ominus`	$\ominus$	`\oplus`	$\oplus$
`\oslash`	$\oslash$	`\otimes`	$\otimes$
`\pm`	$\pm$	`\setminus`	$\setminus$
`\sqcap`	$\sqcap$	`\sqcup`	$\sqcup$
`\star`	$\star$	`\times`	$\times$
`\triangleleft`	$\triangleleft$	`\triangleright`	$\triangleright$
`\uplus`	$\uplus$	`\vee`	$\vee$
`\wedge`	$\wedge$	`\wr`	$\wr$

Symbols: arrows

`\downarrow`	$\downarrow$
`\Downarrow`	$\Downarrow$
`\hookleftarrow`	$\hookleftarrow$
`\hookrightarrow`	$\hookrightarrow$
`\leftarrow` or `\gets`	$\leftarrow$
`\Leftarrow`	$\Leftarrow$
`\leftharpoondown`	$\leftharpoondown$
`\leftharpoonup`	$\leftharpoonup$
`\leftrightarrow`	$\leftrightarrow$
`\Leftrightarrow`	$\Leftrightarrow$

Symbols: arrows...

`\longleftarrow`	$\longleftarrow$
`\Longleftarrow`	$\Longleftarrow$
`\longleftrightarrow`	$\longleftrightarrow$
`\Longleftrightarrow`	$\Longleftrightarrow$
`\longmapsto`	$\longmapsto$
`\longrightarrow`	$\longrightarrow$
`\Longrightarrow`	$\Longrightarrow$
`\mapsto`	$\mapsto$
`\nearrow`	$\nearrow$
`\nwarrow`	$\nwarrow$

Symbols: arrows...

`\rightarrow` or `\to`	$\rightarrow$
`\Rightarrow`	$\Rightarrow$
`\rightharpoondown`	$\rightharpoondown$
`\rightharpoonup`	$\rightharpoonup$
`\rightleftharpoons`	$\rightleftharpoons$
`\searrow`	$\searrow$
`\swarrow`	$\swarrow$
`\uparrow`	$\uparrow$
`\Uparrow`	$\Uparrow$
`\updownarrow`	$\updownarrow$
`\Updownarrow`	$\Updownarrow$

Symbols: big symbols

\sum	$\sum$	\int	$\int$	\oint	$\oint$
\prod	$\prod$	\coprod	$\coprod$	\bigcap	$\bigcap$
\bigcup	$\bigcup$	\bigsqcup	$\bigsqcup$	\bigvee	$\bigvee$
\bigwedge	$\bigwedge$	\bigodot	$\bigodot$	\bigotimes	$\bigotimes$
\bigoplus	$\bigoplus$	\biguplus	$\biguplus$		

LaTeX Course 2011

Part 5: Graphics

Arho Virkki

VTT Technical Research Centre of Finland

Adding images

Adding images to text became a routine task around the beginning of 1990's.[1]

Because of this, the commands to handle graphics are loaded separately to LaTeX even today:

```
\usepackage{graphicx}
\usepackage{color}
```

or

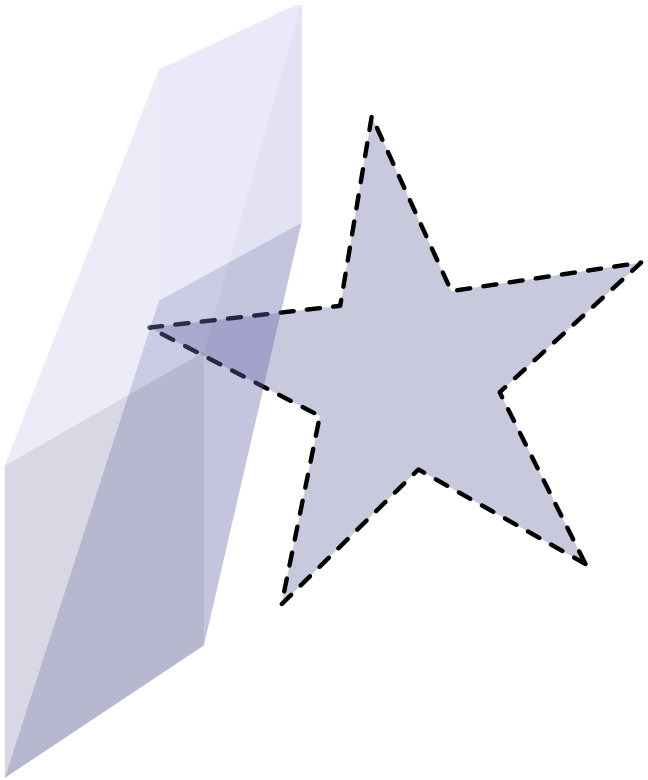
```
\usepackage{graphicx,color}
```

[1] At least to what the author can remember from those early days of computing e.g. without stuff like Internet.

Adding images...

A simple example:

```
\includegraphics{img/foo}
```



Adding images...

- `\includegraphics{file_with_no_extension}` adds an image to the document
- The extension of the file (.eps, .pdf, .png, .jpg) was intentionally left out: The compiler (e.g. `latex` or `pdflatex`) will choose the file with the correct extension on its own.
- Writing the extension explicitly will force the selection (and cause an error on certain cases:
  - The traditional LaTeX understands only the `.eps` files
  - PDFLaTeX can read `.pdf`, `.png` and `.jpg` files, but **not** the `.eps` files $\Rightarrow$ brilliant!...

Adding images...

Changing the format between

`.pdf` ↔ `.eps` ↔ `.png` ↔ `.jpg`

is nevertheless easy.

- In unix, the work can be done with the commands `epstopdf` or `pdftops`[2] or using
- the GIMP or ImageMagic programs (freely available for Windows, Linux and Mac).

[2] The conversion between the 2005 and 2011 LaTeX course graphics files (including several eps illustrations) was done in command line with the single loop: `for f in *eps; do epstopdf $f; done`

Adding images...

By default, the image files are searched from the same folder where the manuscript resides. If there are plenty of images, it might be a good option to store them on a separate sub-folder. In this case, the images must be linked with

- the relative path `\includegraphics{image_path/foo}` or
- using the command `\graphicspath{{}{}...{}}`

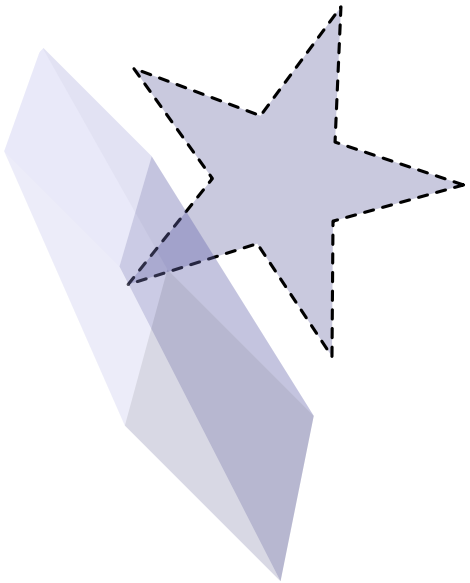
Example:

```
\graphicspath{{/home/arho/images/}
              {images/}}
\includegraphics{images/foo}
```

\includegraphics[]{}

`\includegraphics` accepts quite a number of optional key-value parameters. For example:

```
\includegraphics[width=0.3\textwidth,
                 angle=45]{img/foo}
```



The most common optional parameters include:

- `scale=number` scaling the image as compared to the original
- `width=length` setting the width
- `height=lenght` setting the height
- `angle=degrees` turning the image counterclockwise (=to the positive direction in the mathematical sense)

Note! The lenght must incorporate the unit, e.g. `1.5cm` or `0.2\textwidth` and the scaling factors are with no units.

Figure environments

Sometimes, adding the images to the proper position can be left to the computer, and it is always a good idea to let the machine to keep record of the actual image number. Example:

```
\begin{figure}[!htbp]
\begin{center}
\includegraphics[width=0.4\textwidth]{img/foo}
\caption{A schematic illustration of the Gadget}
\label{fg:foo}
\end{center}
\end{figure}
```

Figure environments...

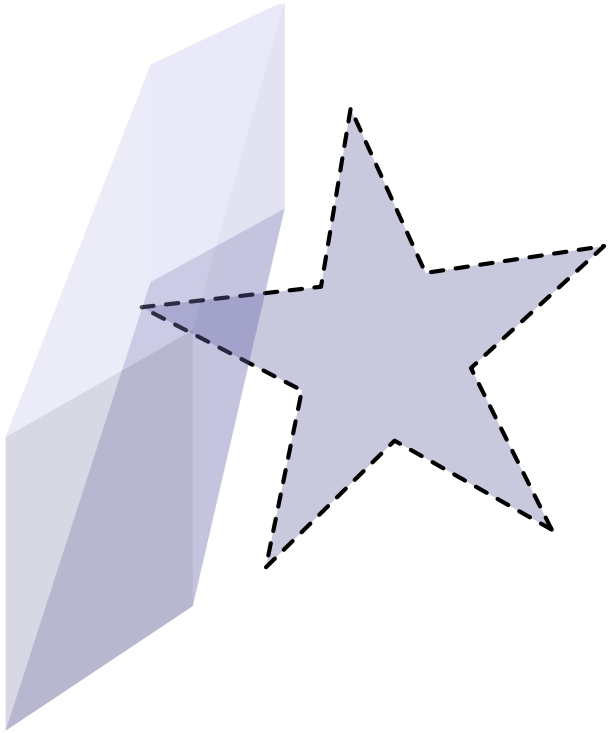


Figure: A schematic illustration of the Gadget

The image can be cited later with `\ref{fg:foo}` (which will be replaced with the actual image number in the compiled document).

Figure environments...

The optional parameter `[!htbp]` means, that

- I really would like(**!**) to insert
- the picture right **h**ere or, if not,
- to the **t**op or
- to the **b**ottom of this page and,
- if this also fails, to a separate **p**icture page.

Figure environments...

There exists an additional package `floatflt` for making the text to surround the images[3]

```
\begin{floatingfigure}{4cm}
\includegraphics[width=3cm]{img/foo}
\caption{StarBox3D}
\label{fg:fooii}
\end{floatingfigure}
By inspecting the figure, we see that the optimal
decision threshold for separating the control and
positive classes is obviously at $b$, where also
the conditional probability $p(P|b)=0.5$. However,...
```

[3] Apparenty there are some issues with the freedom of the license of this package, but it can be installed by hand

http://blogs.fau.de/johanneshabich/2010/05/20/latex-floatflt-sty-missing-on-ubuntu-lucid-10-04/

Figure environments...

By inspecting the figure,
we see that the optimal decision threshold for separating the control and positive classes is obviously at b, where also the conditional probability $p(P|b) = 0.5$. However,...

Figure: StarBox3D

Colors

Colors are easiest to add with the package

```
\usepackage{color}
```

but the number of colors is rather limited — but should at least include white, black, red, green, blue, cyan, magenta and yellow[4]. To define a custom color, we need the additional command

```
\definecolor{col_name}{col_system}{col_def}
```

For example

```
\definecolor{mygreen}{rgb}{0.61,0.78,0.05}
\colorbox{vihrea}{Nice Green Backround!}
```

Nice Green Backround!

[4] http://en.wikibooks.org/wiki/LaTeX/Colors

Colors...

Using the pre-defined colors is easy

```
\colorbox{orange}{Orange}
\fcolorbox{red}{green}{Pearl}
\textcolor{blue}{
\[ \sum_{k=1}^\infty a_k \]}
```

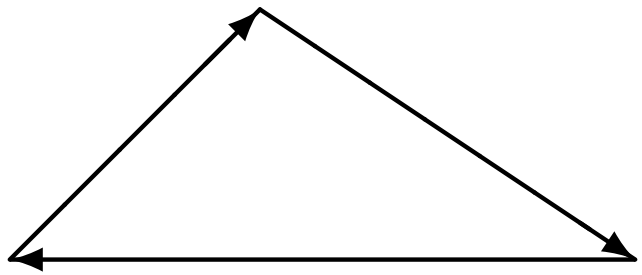
Orange Pearl

$$\sum_{k=1}^{\infty} a_k$$

Using LaTeX to draw images

LaTeX has a built-in system to draw simple images. The following example is from the book of Kopka and Daly.

```
\setlength{\unitlength}{0.8cm}
\begin{picture}(5,2)\thicklines
  \put(5,0){\vector(-1,0){5}}
  \put(0,0){\vector(1,1){2}}
  \put(2,2){\vector(3,-2){3}}
\end{picture}
```



Drawing images...

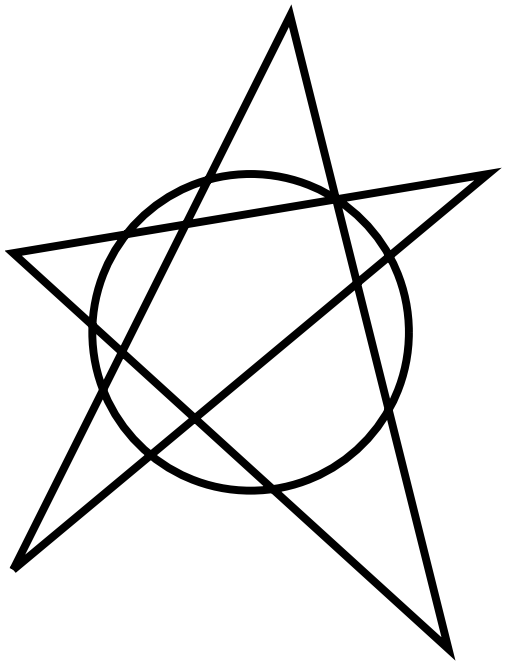
For a comparison, some PostScript code[5] (again from K&D):

```
%!PS-Adobe-3.0 EPSF-3.0
%%BoundingBox: 169 158 233 242
220 200 moveto
200 200 20 0 360 arc
170 170 moveto
230 220 lineto
170 210 lineto
225 160 lineto
205 240 lineto
170 170 lineto
stroke
showpage
```

[5] PostScript is nowadays superseded by the pdf format

Drawing images...

This code produces the image

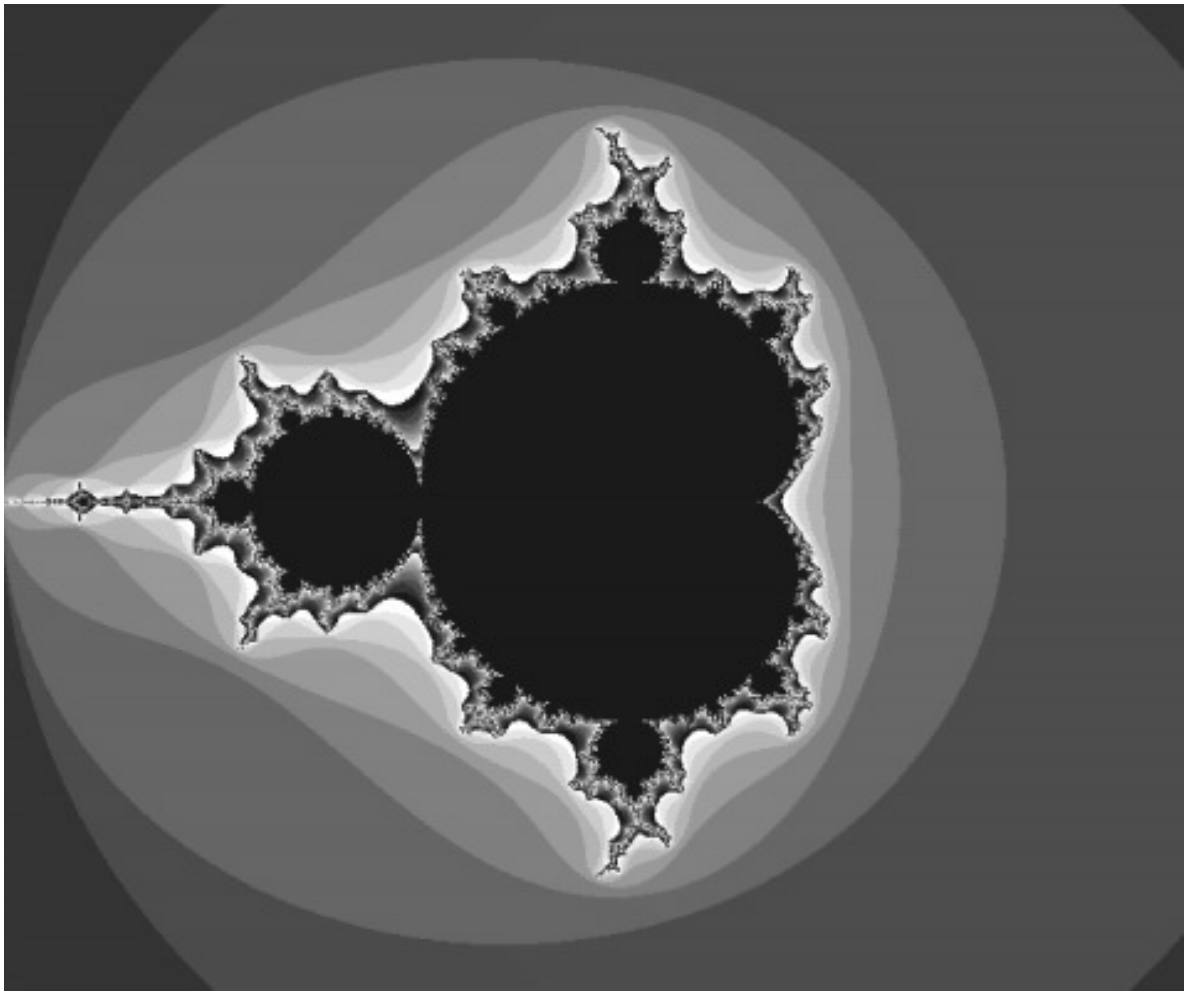


The following example is a bit more complicated. . .

Drawing images...

```
%!ps
/iter 60 def /reso .005 def /sq { dup mul } def
/mod { 2 copy div floor mul sub } def /plot {
newpath moveto 1 0 rlineto stroke } def gsave
280 420 translate 260 2 div dup scale 2 260 div
setlinewidth -2 reso 2 { /x exch def -2 reso 2 {
/y exch def /r 0 def /i 0 def /n 0 def iter { r
sq i sq add 4 gt { exit } if /rr r sq i sq sub x
add def /i 2 r mul i mul y add def /r rr def /n
n 1 add def } repeat n 10 mod .1 mul .1 add
setgray x y plot } for } for grestore showpage
```

Drawing images...



Drawing images...

- The command languages are precise, because the images are not stored as pixels, but as lines, circles, Bézier curves,. . . , and text (with the corresponding size and font).
- For some purposes, using svg or similar language might be beneficial.
- The LaTeX default language is rather poor, as e.g. the angles of the line can only have some fixed values
- A better investment for a casual used is to learn a drawing tool (e.g. Inkscape[6] for vector graphics or Gimp[7] for digital photographs)

[6] http://inkscape.org/
[7] http://www.gimp.org

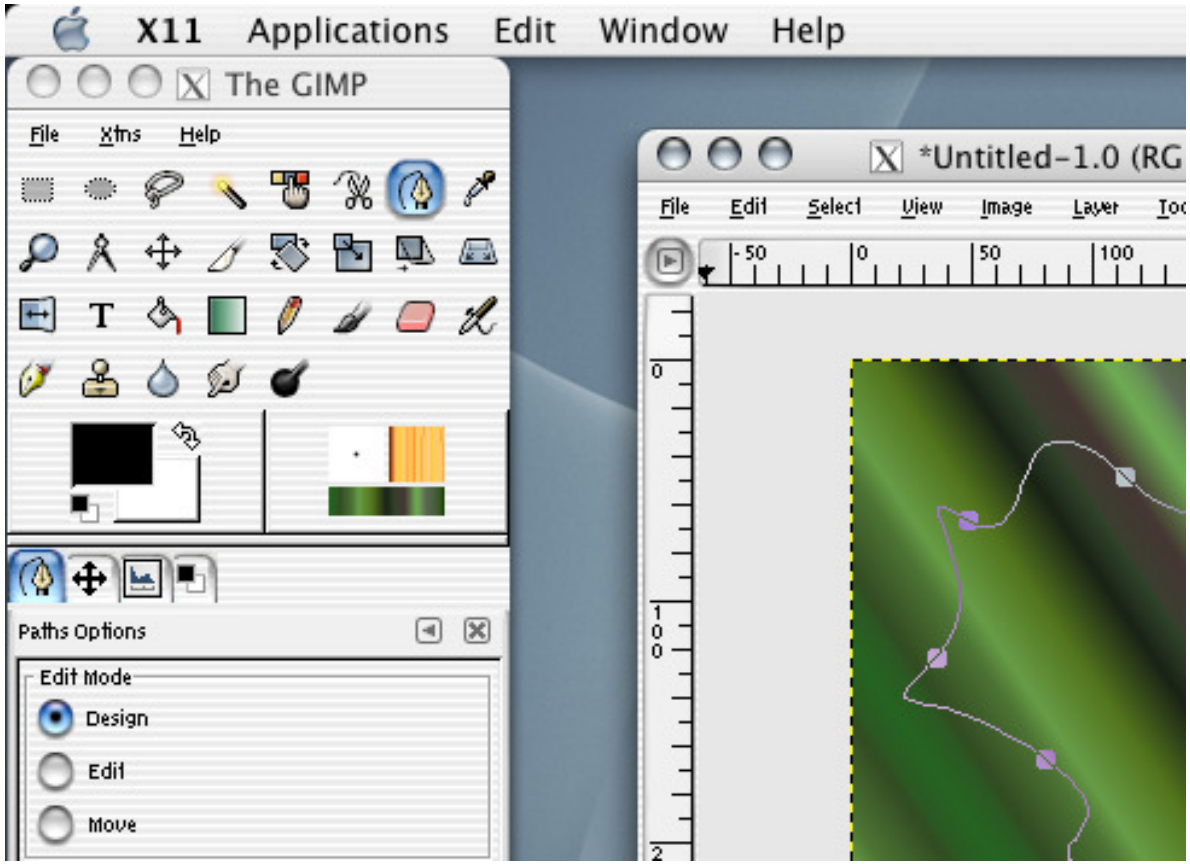
Drawing images...

Which program to use? The answer depend on the personal taste (as well as the willingness to pay for commercial software). Nevertheless, if we restrict ourselves only to the free options,

- GIMP is the choice for painting (pixel map images like digital photos)
- Inkscape is the choice for vector graphics (where e.g. the circles are truly Platonian, and the device (= printer or screen) only does its best to approximate the idea)
- Programming environments like R, Matlab and gnuplot all produce high-quality plots in pdf format (which can be further polished with Inkscape, if desired).

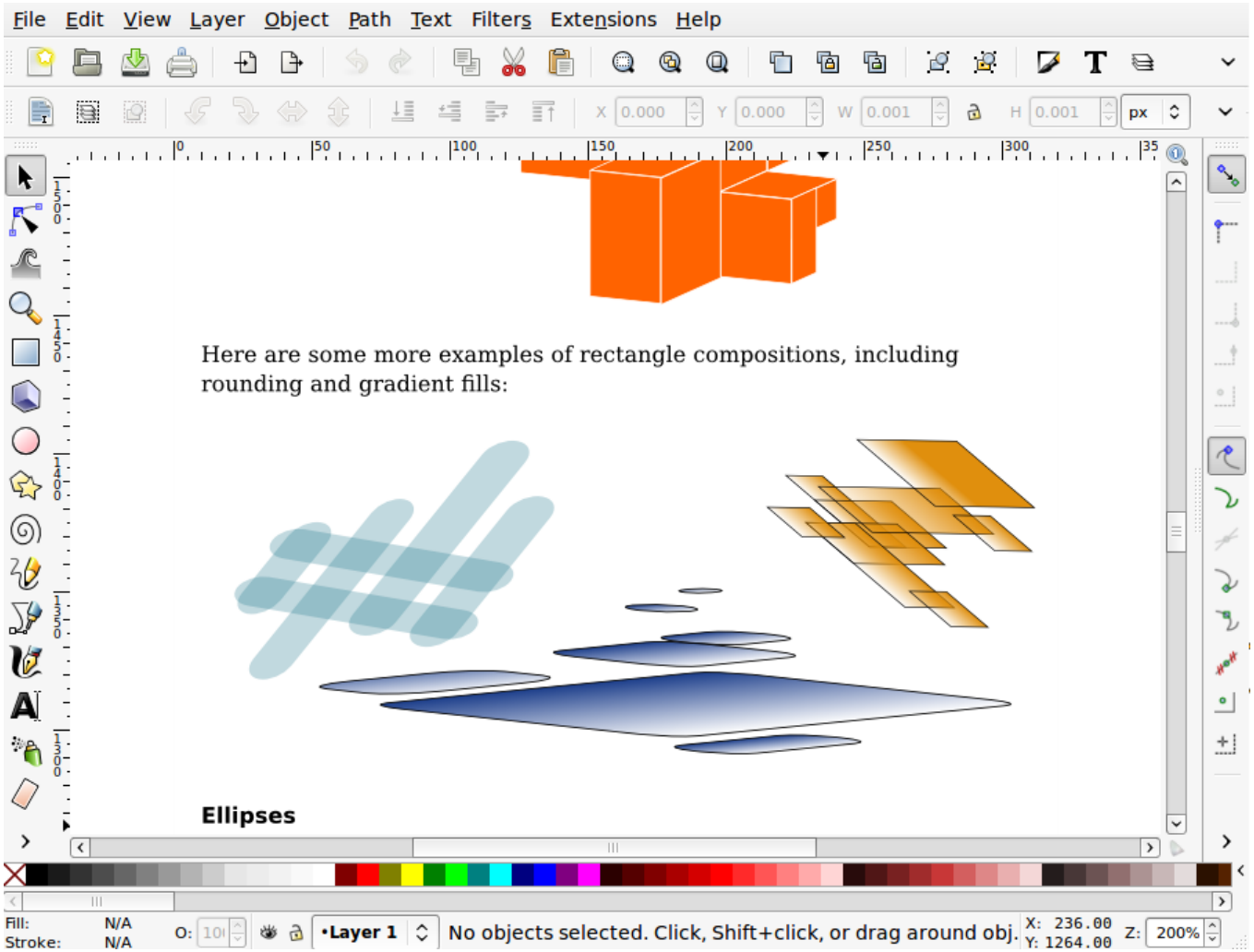
Drawing images...

Gimp is purely a program for painting (bitmap images, including digital photographs and scanned text pages)



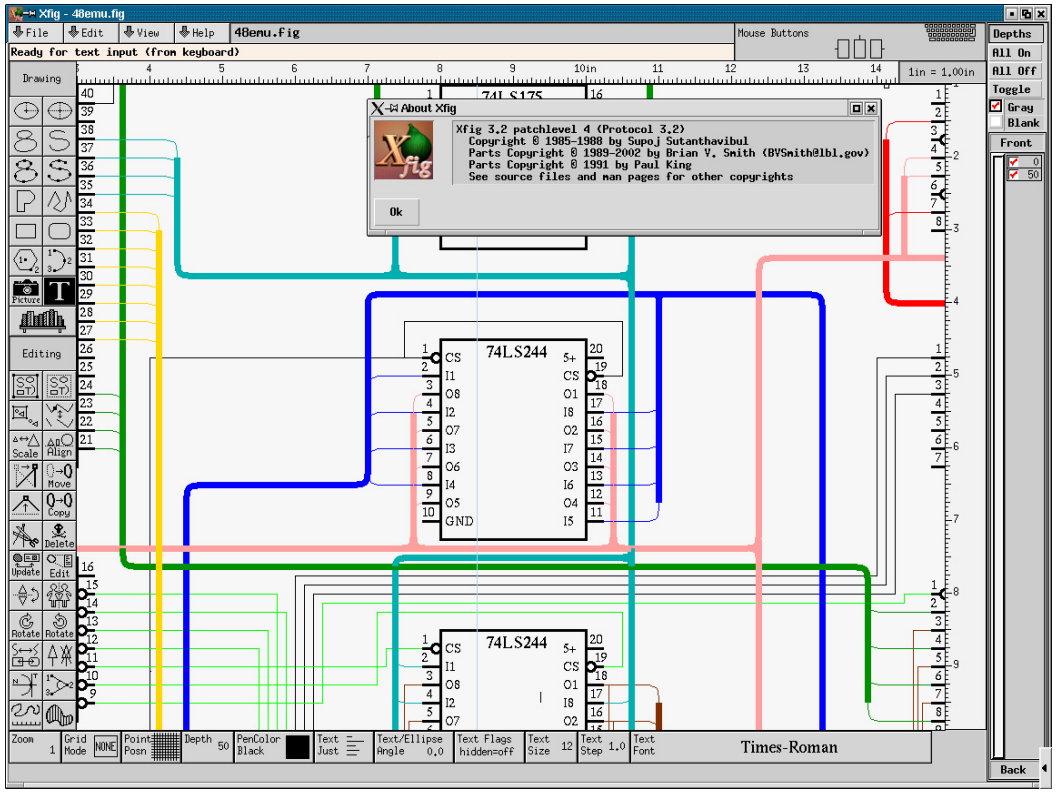
Drawing images...

Inkscape is for vector graphics and resembles the CorelDraw! software (originated at the early 90's)



Drawing images...

Xfig has not been much updated since 2002 (and the latest change is from 2007) but some people are so used to it, that the program can still be found e.g. from the default Debian repositories (and has also been ported to Mac OS X).



Drawing images...

`gnuplot` is yet another “traditional tool”[8] which uses its own command language[9]

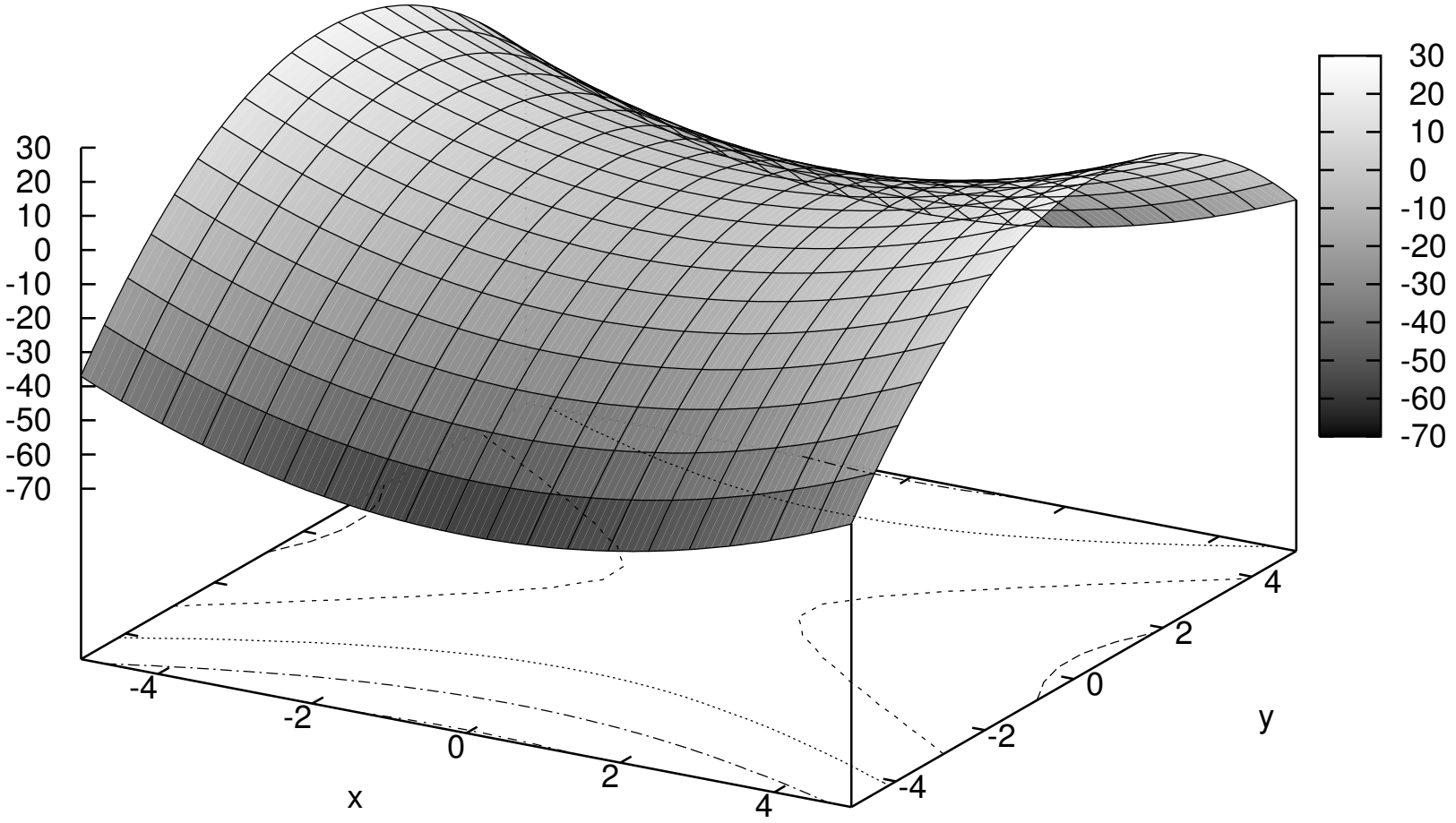
```
set pm3d
set contour base
set xrange [-5:5]
set yrange [-5:5]
set isosamples 20,20
set xlabel "x"
set ylabel "y"
unset key
set term post eps enhanced
set output "gnuplotex.eps"
splot x**2-2*y**2 + 2*y -2
```

[8] Observe that all these “obsolete” tools are younger than LaTeX...

[9] `set pm3d` refers to the OS/2 presentation manager.

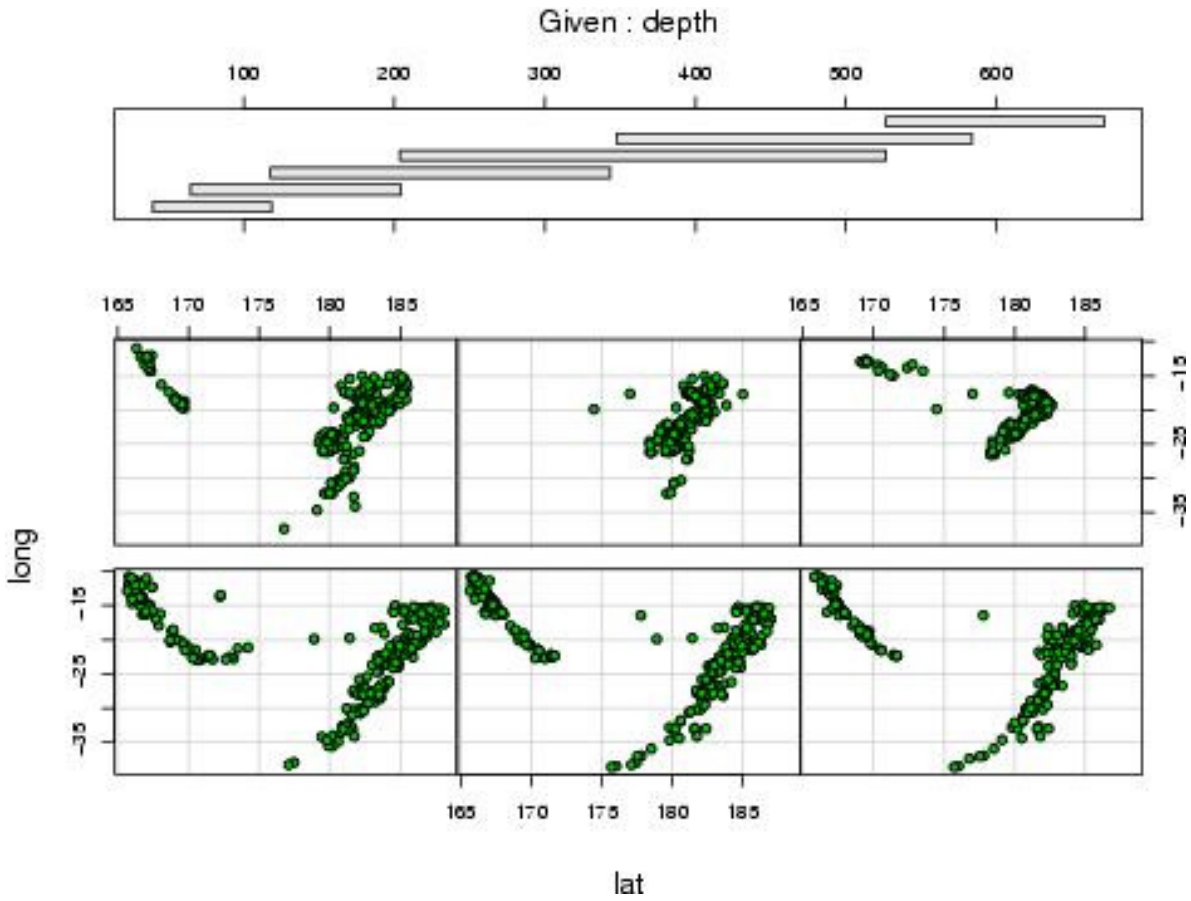
Drawing images...

The previous code produces the image



Drawing images...

The R language and environment for statistical computing[10] is an excellent choice for drawing statistical images



[10] http://www.r-project.org/

Drawing images...

Of course, there are commercial options for vector graphics:

- CorelDraw!,
- Adobe Photoshop,
- Adobe Illustrator,...

and software for mathematical illustrations:

- Maple,
- Mathematica,
- Matlab,...

Inkacape use case: Images with equations

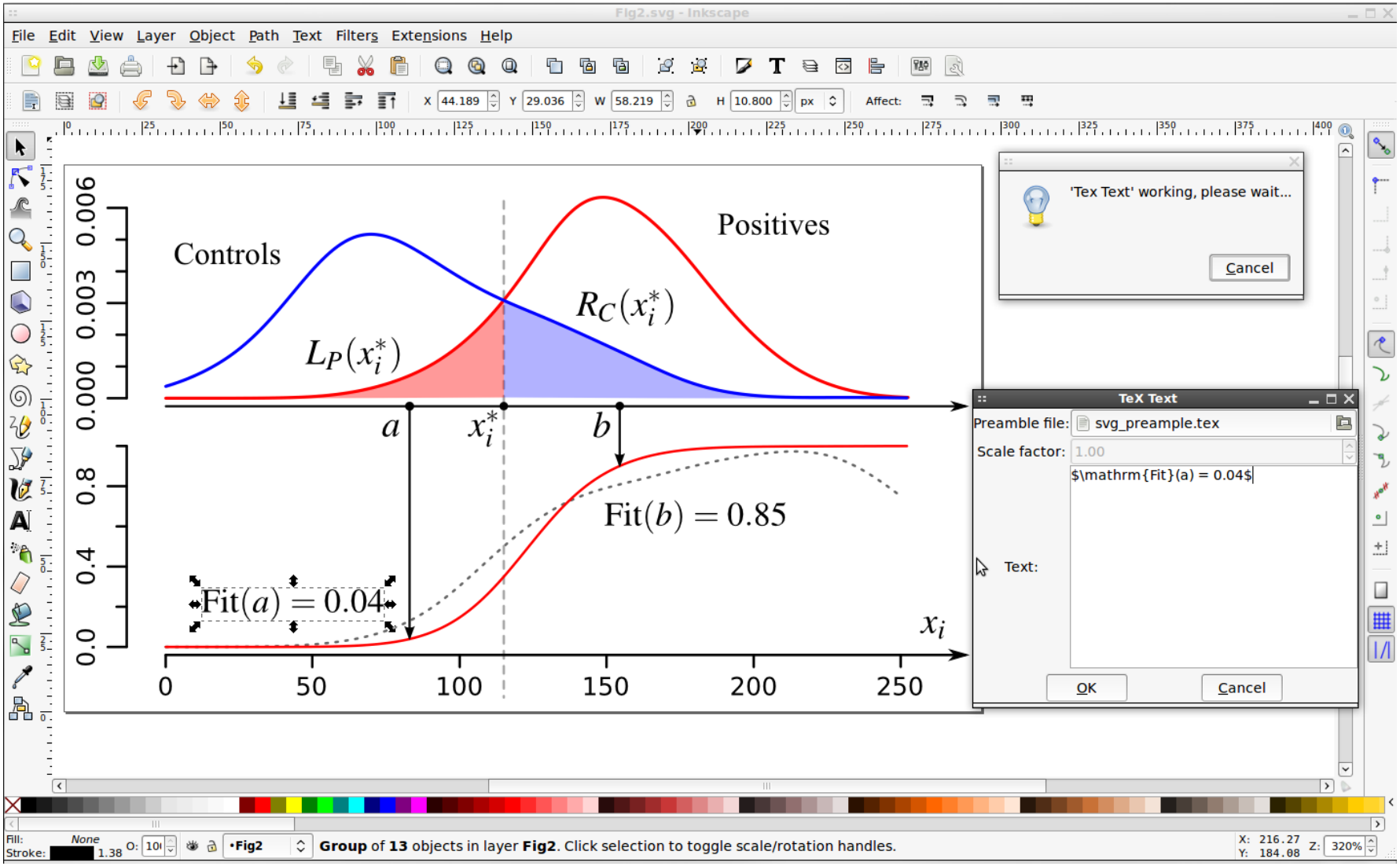
Suppose that you are producing an A0-sized poster presentation for a conference and would e like to include text and images with equations embedded. The two feasible options are:

- Use Inkscape to produce the whole thing or
- Use whichever software you like for the presentation and the images, but polish the separate figures using Inkscape

with Pauli Virtanen's `textext` plug-in installed from http://pav.iki.fi/software/textext/[11]

[11] The installation instructions are non-trivial and different for each platform, but the for superior quality, the work is worth of it. . .

Inkscape use case: Images with equations...



Inkacape use case: Images with equations. . .

The final result in: "J Mattila, J Koikkalainen, A Virkki, M van Gils, and J Lotjonen. Design and application of a generic clinical decision support system for multi-scale data. Transactions on Biomedical Engineering, (in press), 2011"[12]

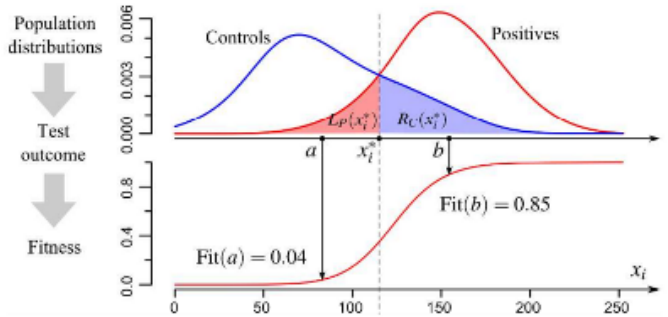


Figure 1. Probability density functions of C_i and P_i, the resulting *fitness* (with examples at test outcome values a and b), and the optimal classification threshold x_i*.

The software library uses a statistical approach to analyze multi-scale data and combine them into an aggregate representation interpretable by a clinician. It supports heterogeneous patient data of virtually any type and scale and allows clinicians to study the system simultaneously as a collection of components and as a whole. The library has been designed to easily support several diseases, requiring minimal amount of configuration. The first application prototype developed using the proposed decision support library is a CDSS tool for early diagnosis of AD. The statistical methods are validated using data from several medical datasets and the clinical applicability of our proposed system is demonstrated by evaluating the implementation of the CDSS tool.

The main contributions of this work are the description of

$$DSI(x_1, x_2, \cdots, x_n) := \frac{\sum_{i=1}^{n} Rel(i) Fit(x_i)}{\sum_{i=1}^{n} Rel(i)}, \quad (1)$$

where $Rel(i)$ is a *relevance* function providing the weighting between [0,1] for variable i and $Fit(x_i)$ is a *fitness* function providing a non-linear transformation of value x_i into *fitness* space [0,1].

A *fitness* function computes the location, i.e. rank, of an individual variable x_i relative to values of the same variable in two different populations, denoted as controls C_i and positives P_i. Our system currently supports scalar, ordinal, and categorical (including boolean) variables, but could be extended to support others, such as value lists and complex values, by deriving appropriate *fitness* functions. Let us consider a scalar variable where the progression of a disease tends to increase its value (see Figure 1). For these, *fitness* is defined as a monotonically increasing function

$$Fit(x_i) := \frac{L_P(x_i)}{L_P(x_i) + R_C(x_i)}, \quad (2)$$

where $L_P(x_i)$ is the left integral of probability density function (PDF) for positive class values P_i and $R_C(x_i)$ is the right integral of PDF for control class values C_i. Derivation of the *fitness* function can be conducted in an analogous manner for ordinal variables. For a categorical variable $x_i \in \{\Omega_1, \cdots, \Omega_n\}$, we use as *fitness* the conditional probability of the subject belonging to the positive population in the case of observing $\Omega = x_i$.

[12] http://www.vtt.fi/aivotutkimus

LaTeX Course 2011

Part 6: Companion tools

Arho Virkki

VTT Technical Research Centre of Finland

The need for extra tools

As seen earlier, LaTeX is a capable system for typesetting. For productive work, we still need

- citation management system,
- advanced drawing tools,
- perhaps some more exotic LaTeX add-on packages, and
- a proper version control system (e.g. cvs, svn or git)[1]

[1] These tools are not discussed in this course. Nevertheless, LaTeX users get their full power, as the manuscripts are text – similar to software source code.

BibTeX

BibTeX is the de facto standard for LaTeX citatioin management (written by Oren Patashnik and Leslie Lamport in 1985). It has

- An uncomplicated structure:
  - the database can be modified by hand and
  - it is easy to write programs that can handle BibTeX files.
- A lot of BibTeX tools already exists, including
  - `bibtool`
  - `bibtex2html`
  - `...`

BibTeX...

BibTeX-database is just a plain text file. Example:

```
@Article{gro67,
  author  = {Fred S. Grodins and ...},
  title   = {Mathematical analysis and ...},
  journal = {Journal of Applied Physiology},
  volume  = {22},
  number  = {2},
  pages   = {260-276},
  year    = {1967},
  url     = {Grodins - Maadsotrcs.pdf}
}
```

BibTeX...

- The record begins with the type, for example `@article`, `@book` or `@techreport`.
- The first string is the key for the citation, which can be used in the text as `\cite{lyhenne}`.
- All fields are key-value pairs, like `pages={260-276}`.
- All fields are separated by commas.

The format of the record is hence

```
@tyyppi{field, field, ..., field}
```

BibTeX...

- The authors are introduced in the order *first_name last_name* and
- the names are separated with keyword `and`.

For example,

```
J. Doe and O. Normalverbraucher and
M. Meikalainen
```

The style of the Bibliography is defined by a separate style sheet. This can usually be downloaded from the home page of the journal the article is being submitted.

BibTeX...

The style sheet can e.g. define that the title begins with a capital letter, and all the following words are written in lower case. In this case, for example,

```
Stability of the Human Respiratory
Control System I: Analysis of a
Two-Dimensional Delay State-Space Model
```

becomes

```
Stability of the human respiratory
control system i: analysis of a
two-dimensional delay state-space model
```

which was not necessarily the original intention.

BibTeX...

If we need to fix the presentation of a word, that part of the text can be written inside extra curly braces { }

```
... System {I}: {A}nalysis of a ...
```

which makes these parts immutable.

BibTeX and natbib

Using the add-on package

```
\usepackage{natbib}
```

we get two additional commands in addition to the basic `\cite{}`:

- `\citet{}` — the *textual* and
- `\citep{}` — the *parenthetical citation*.

that enable taking the citations as part of the text.

BibTeX and natbib...

Example:

```
\citet{Foo88} showed that... furthermore ...
in the last decade \citep{Bar93,Baz96}.
```

produces

> *Fooman et al. (1988) showed that... furthermore ... in the last decade (Barnos et al., 1993; Bazel et al., 1996).*

depending of the chosen style.

BibTeX usage

Example: If BibTeX database is stored in the file

`citations.bib`, it can be cited in the text using the command `\cite{key}`, provided that the file has been added to the list of citation files:

```
\bibliographystyle{apalike}
\bibliography{citations}
```

where `apalike.bst` is one of the default style sheets available.

BibTeXin usage...

In case that the document (here `text.tex`) is compiled for the first time, one needs to run all the commands

```
latex text
bibtex text
latex text
latex text
```

until all is done.

Style sheet tricks

One of our very first examples had the command

```
@Article{gro67,
  ...,
  url      = {Grodins - Maadsotrcs.pdf}
}
```

where `url` refers to the file name in the local machine. This, of course, has no meaning for other users, and the field is best to be removed from the public version of the file.

Style sheet tricks...

Because the `.bib` files are just ordinary text, removing a field is easy.

Example: (`sed` = strean editor)

```
sed --in-place '/url/ d' citations.bib
```

The same can also be done with a proper Python or R program.

Beamer

Beamer is an additional package for writing slides.

- You write ordinary LaTeXand
- Prosper makes to output to look like it was produced using “the power point method”.
- — except for the formulas, which will still be typeset with the LaTeXquality.
- The style definitions of the current slide set are

```
\usetheme{Warsaw}
\usecolortheme{seahorse}
\usefonttheme{professionalfonts}
```

Beamer...

Beamer comes installed by defauld with LiveTeX, and compiling the slides (in `slides.tex`) is straightforward:

```
pdflatex slides
```

- You had better to choose a program that automatically notices the update `pdf`-file.
- Under linux, `evince` automatically notices any change in the file. For Mac OS X, the editor usually incorporates a proper previewer.

Beamer example

The following commands tell LaTeX that the document is a slide set (and sets some extra preferences)

```
\documentclass[13pt]{beamer}
\usetheme{Warsaw}
\usecolortheme{seahorse}
\usefonttheme{professionalfonts}
\setbeamertemplate{navigation symbols}{}

\begin{document}
...
```

Beamer example...

Each slide begins and ends with

```
\begin{frame}[fragile]
\frametitle{The slide title}
This goes to the slide
...
\end{frame}
```

The extra argument `[fragile]` is not always necessary, but it enables using the `verbatim` environment.[2]

[2] Which is heavily used e.g. for these slides.

Beamer example...

These slides are actually bad. Good slides have as little text as possible, and

- **the text is large enough and it**
- **has lots of free space around it.**

In addition,

- *each slide contains at most 4 key elements, and*
- *the audience is not expected to read and listen at the same time.*