DTU Civil Engineering
Department of Civil Engineering

Subject-Driven Generation Techniques for Stable Diffusion Model

Thesis subtitle

Master Thesis

Subject-Driven Generation Techniques for Stable Diffusion Model
Thesis subtitle

Master Thesis
June, 2023

By
Mario Lozano Cortés

Cover photo: Vibeke Hempler, 2012
Published by: DTU, Department of Civil Engineering, Brovej, Building 118, 2800 Kgs. Lyngby Denmark
www.byg.dtu.dk
ISSN: [0000-0000] (electronic version)
ISBN: [000-00-0000-000-0] (electronic version)

ISSN: [0000-0000] (printed version)
ISBN: [000-00-0000-000-0] (printed version)

Approval

This thesis has been prepared over six months at the Section for Indoor Climate, Department of Civil Engineering, at the Technical University of Denmark, DTU, in partial fulfilment for the degree Master of Science in Engineering, MSc Eng.

It is assumed that the reader has a basic knowledge in the areas of statistics.

Mario Lozano Cortés - s226536

.....

Signature

.....

Date

Abstract

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Acknowledgements

Mario Lozano Cortés, MSc Civil Engineering, DTU
Creator of this thesis template.

[Name], [Title], [affiliation]
[text]

[Name], [Title], [affiliation]
[text]

Contents

1 Introduction

This template complies with the DTU Design Guide https://www.designguide.dtu.dk/. DTU holds all rights to the design programme including all copyrights. It is intended for two-sided printing. The `\cleardoublepage` command can be used to ensure that new sections and the table of contents begins on a right hand page. The back page always ends as an odd page.

All document settings have been gathered in Setup/Settings.tex. These are global settings meaning the settings will affect the whole document. Defining the title for example will change the title on the front page, the copyright page and the footer. A watermark can be enabled or disabled in Setup/Premeable.tex. You can edit the watermark to display draft, review, approved, confidential or anything else. By default the watermark is printed on top of the contents of the document and has a transparent grey colour.

1.1 This is a section

Every chapter is numbered and the sections inherit the chapter number followed by a dot and a section number. Figures, equations, tables, ect. also inherit the chapter numbering.

1.1.1 This is a sub section

Sub sections are also numbered. In general try not to use a deep hierarchy of sub sections (`\paragraph{}` and the like). The document will become segmented which will make the document appear less coherent.

This is a sub sub section

And those are not numbered. It is possible to adjust how deep hierarchy of numbering sections goes in Setup/Settings.tex.

The front and back cover have been made to replicate the examples in the design guide https://www.designguide.dtu.dk/#stnd-printmedia. The name of department heading is omitted because it is located in the top right corner (no need to write it twice). Take a look at https://www.inside.dtu.dk/en/medarbejder/om-dtu-campus-og-bygninger/kommunikation-og-design/skabeloner/rapporter if you want to make your cover separately.

Citing is done with the `biblatex` package [1]. Cross referencing (figures, tables, ect.) is taken care by the `cleveref` package. Just insert the name of the label in `\cref{}` and it will automatically format the cross reference. For example writing the `cleveref` command `\cref{fig:groupedcolumn}` will output “fig. 3.3”. Using `\Cref{}` will capitalise the first letter and `\crefrange{}{}` will make a reference range. An example: Figure 3.2 is an example of a stacked bar chart and figs. 3.1 to 3.3 are three consecutive figures.

1.2 Font and symbols test

Symbols can be written directly in the document meaning there is no need for special commands to write special characters. I love to write special characters like æøå inside my TEX document. Also á, à, ü, û, ë, ê, î, ï could be nice. So what about the “¿” character. What about ° é ® † ¥ ü | œ ‘ @ ö ä ¬ ‹ « © ƒ ß ª … ç ñ µ , · ¡ “ £ ™ [ ] ’. Some dashes - – —, and the latex form - – —

This is a font test
Arial Regular
Arial Italic
Arial Bold
Arial Bold Italic

2 State of the art

Text-to-image is an emerging field of deep learning where models can generate lifelike and highly detailed images from textual descriptions. The development of these models is a challenging task that requires the close integration of both computer vision and NLP approaches. The latest advancements in text-to-image models have led to the capability of producing high-quality images with rich semantic content that can now be used for tons of applications including video games and virtual reality, e-commerce, or education among others. Even though recent advancements have allowed the use for commercial applications, generative models remain a challenging and tough problem. This section aims to analyse the current state-of-the-art of text-to-image models

2.1 Historical review of text-to-image models

2.2 Diffusion probabilistic models

Throughout 2022, the capabilities and popularity of text-to-image models have exploded. The general public is aware of some models, such as DALL-E 2, Midjourney, or Stable Diffusion. Nonetheless, the vast majority of people are unaware of the technical prowess required in the field of Artificial Intelligence for these models to exist. This section aims to shed some light on the internal functioning and processes of these models from an academic perspective.

Diffusion probabilistic models are a class of latent variable models that introduce the ideas of nonequilibrium thermodynamics into data generation techniques by homogeneously adding noise into samples. Thus, they join the list of models that manage to generate high-quality images such as variational autoencoders (VAEs) or Generative adversarial networks (GANs). The latter models have been the reference of academic research in recent years and are the benchmark to be surpassed by diffusion models.

GANs were introduced in 2014 by researchers at the University of Montreal in the paper *Generative Adversarial Nets* [2]. The idea is to create generative models through an adversarial process in which two neural networks compete against each other. One of the networks will be generative while the other will be discriminative. Thus, the generative network will be in charge of capturing the distribution of the training dataset while the discriminative network must distinguish whether a sample comes from the generative network or the training data. The idea is that the generative network maximises the probability that the discriminative network makes errors.

Diffusion models, on the other hand, achieve high-quality image synthesis results in the paper *Denoising Diffusion Probabilistic Models* [3] by researchers from the University of California, Berkeley. These models are based on creating a Markov chain in which at each step they add Gaussian noise to an image in a diffusion process and then learn to undo it. In this way, a network is trained that is capable of reconstructing images from random noise. The differences between GANs and diffusion models are presented in figure 2.1.

Diving further into the workings of diffusion models, we define the **forward process** of the Markov chain. The first step is to take a sample of the target data distribution, which we will call X_0, and add Gaussian noise in T steps. The forward process is thus defined as a Markov chain in which the state of a sample at time n depends only on the state at time $n-1$. Therefore, one can denote the distribution of any sample conditioned on the initial state X_0.

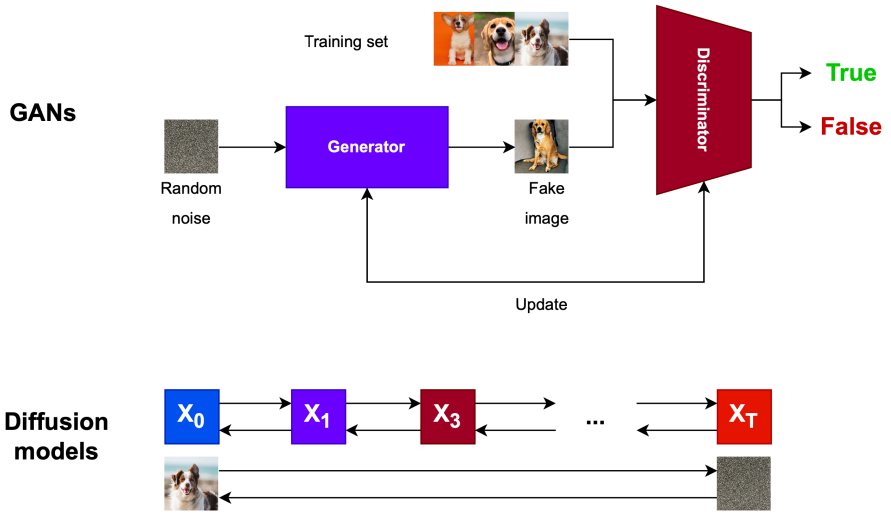

Figure 2.1: Overview of GANs and diffusion models

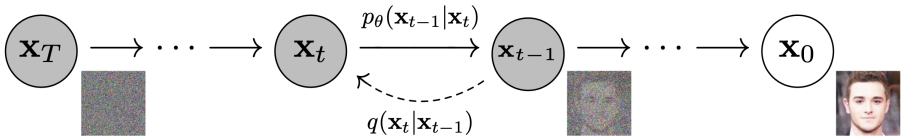

Figure 2.2: Markov chain of the diffusion process [3]

$$q\left(x_{1:T}|x_0\right) = \prod_{t=1}^{T} q\left(x_t|x_{t-1}\right)$$

In every step of the noising process Gaussian noise is added according to some variance schedule $\beta_1 ... \beta_t$, normally consider as hyperparameters. The restrictions applied to β_t are $\beta_1 < \beta_2 ... < \beta_t$ and $\beta_t \epsilon (0, 1)$. I stands for identity.

$$q\left(x_t|x_{t-1}\right) = \mathcal{N}\left(x_t; \sqrt{1-\beta_t x_{t-1}}, \beta_t I\right)$$

As β_t grows in time and T approaches the limit ($T \to \infty$, $\beta_t \to 0$), the Gaussian mean will approach zero with identity covariance. In this way, the distribution will lose all the information contained in the original image. In practice, researchers use a T close to 1000 [3].

$$q\left(x_t|x_0\right) \approx \mathcal{N}\left(0, I\right)$$

Figure 2.2 shows the diffusion process described so far.

In summary, it is proven that the forward process destroys the structure of a data distribution step by step. The next challenge is to learn the **reverse diffusion process** in order to generate data that resembles the training distribution from pure Gaussian noise. As with the forward process, the reverse diffusion process can be expressed as a Markov chain where the probability of a sequence of samples can be expressed as the product of conditional probabilities.

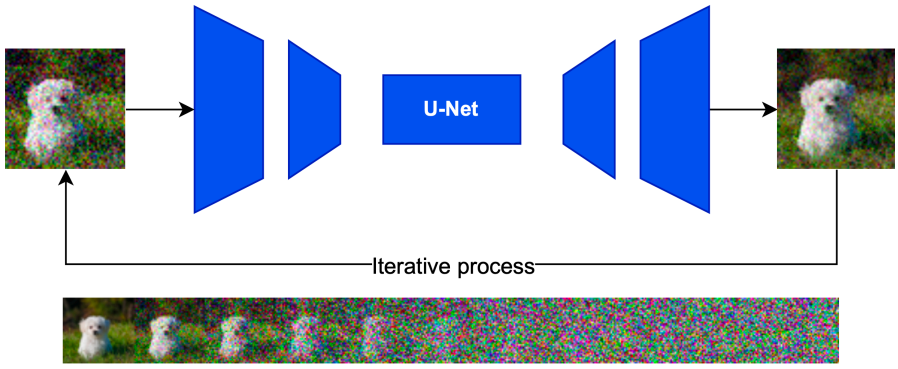

Figure 2.3: Reverse diffusion learning schema

$$p_\theta\left(x_{0:T}\right) := p\left(x_T\right) \prod_{t=1}^{T} p_\theta\left(x_{t-1} | x_t\right)$$

The reverse diffusion process involves a neural network to remove noise from an image in a stepwise manner. Thus, starting from pure Gaussian noise, noise is removed step by step to arrive at an image that resembles the training distribution. The reason that the process has to be done in a stepwise manner is that "*the estimation is more tractable than explicitly describing the full distribution*" as expressed in the publication *Deep Unsupervised Learning using Nonequilibrium Thermodynamics* [4].

The neural network that the authors of DDPM propose aims to **predict the noise to subsequently eliminate it from the image**. This is equivalent to obtaining the mean of the distribution since the authors decide to fix the variance. The authors decide to use the **U-Net network** [5] for this purpose. U-Net consists of a bottleneck in the middle that ensures that the network removes irrelevant information and focuses on the important information. In addition, the network, between the encoder and the decoder, uses residual connections to improve efficiency. Finally, the authors of DDPM decide to employ self-attention at the 16 × 16 feature map resolution. Figure 2.3 shows a schema of the learning process.

Another question that arises when working with diffusion models is how conditional generations can be provided. This can be achieved through various techniques. One way is to feed a conditional variable into the training so that the model makes use of it in the generation to resemble a subset of the training distribution. However, guiding the generation process through a classifier is a more flexible technique that allows even more complex text descriptions than simple labels to be worked with. The idea is to take an already trained classifier and **guide the generation in the direction of the gradient of the classifier label**.

2.2.1 Improvements to diffusion probabilistic models

2.3 Latent diffusion models

2.4 Stable diffusion

2.4.1 Training dataset

A significant challenge posed by models like Stable Diffusion is the choice of images they are trained on. This issue is not insignificant as image generation models require both textual descriptions of training images and a sufficient amount of variety to enable the model to comprehend how the world is constructed and thus be capable of reproducing it. However, the conventional datasets of the Machine Learning field (COCO, ImageNet, etc) fail to satisfy these

requirements since they are not intended for this purpose. Researchers have discovered that the solution is the web, where a vast array of diverse images about the world can be found, many of which have HTML alt attribute tags.

Stable Diffusion has an advantage over some of its rivals, including *DALL-E 2*, in that it is an open-source project, meaning that the dataset employed for training is well-known and accessible to everyone. Specifically, the dataset used by Stable Diffusion is "***LAION-5B**, a dataset of 5.85 billion CLIP-filtered image-text pairs, 14x larger than **LAION-400M**, previously the biggest openly accessible image-text dataset in the world*" [6]. In particular, Stable Diffusion presents several checkpoints on various LAION-5B assemblies. Some of these checkpoints in Stable Diffusion version 1 [7] are:

- **stable-diffusion-v1-1**: 256 x 256 images from a subset of 2.3 billion English-captioned images called **LAION-2B-EN**.
- **stable-diffusion-v1-2**: Resumed training on *stable-diffusion-v1-1* with 512x512 images from the subset **LAION-2B-EN**, containing a selection of improved aesthetics images compared to the others.
- **stable-diffusion-v1-3**: Resumed training on *stable-diffusion-v1-2* with the same subset of images but a 10% dropping of the text-conditioning.
- **stable-diffusion-v1-4**: Resumed training on *stable-diffusion-v1-2* with 512x512 images from the subset **LAION-Aesthetics v2 5+**, containing 600 million images from **LAION-2B-EN** with better aesthetics and low-resolution and watermarked images filtered out.
- **stable-diffusion-v1-5**: *stable-diffusion-v1-4* trained with more steps.

LAION-5B retrieves images from the internet that are not uniformly high in quality. Because these images are gathered automatically, they do not adhere to the same rigorous standards as other image datasets. As a result, the checkpoints for training Stable Diffusion use varying subsets of LAION-5B. Nonetheless, the fact that the images obtained are not accurately labelled as they are in standard vision supervised learning is actually an advantage. Consequently, Stable Diffusion is now included in the group of architectures, such as CLIP or DALL-E 2, that have proven the value of these vast datasets, even though they contain a significant amount of noise.

LAION-5B contains 5.85 billion image-text pairs divided into three subsets. **LAION2B-EN**, which contains 2.32 billion English image-text pairs; **LAION2B-MULTI** with 2.26 billion image-text pairs from all other languages (Russian, French and German as top 3) and **LAION1B-NOLANG** of 1.27 billion samples where the language is not correctly defined.

LAION-5B Description

Attribute	Description
id	Image identifier
URL	URL from where the image was obtained
Text string	Caption accompanying the image
Dimensions	Height and width of the image
Similarity	Cosine similarity between the text and image embeddings. CLIP-based models are employed to gauge the level of accuracy with which an image is described by a given textual description.
pwatermark	Probability that the image presents a watermark. The value is obtained by a custom model trained by LAION. Value between 0 and 1
punsafe	Probability that the image is NSFW. As some of the content acquired from the web may not be suitable for all audiences, LAION employs a custom model to assess its adequateness. Value between 0 and 1

3 Examples of figures, tables, equations and listings

In the following a bunch of examples of figures and tables have been made. There are advantages to using `tikZ` diagrams over excel diagrams. 1) the font and font size perfectly matches the document 2) the styling and colours are pre-defined to follow the design guide 3) the plots uses vector graphics which reduces the file size, reduces the compile time and looks sharp when zooming in. The possibilities are endless, look at the `pgfplots` gallery for inspiration: http://pgfplots.sourceforge.net/gallery.html. However there are still cases where I would recommend to insert a plot as a picture. For example if the plot contains a lot of data: a line graph with 1000 points takes a long time to compile.

Some tips if you want good looking diagrams or graphs which will be inserted as pictures (e.g. in a figure environment with \includegraphics): The main font is Arial. Use DTU colours as described in chapter 2. Use high quality pictures. Try to scale the diagram (picture) so the text size of the axis legends match the text size in this document.

Remember to change the label of your figures so there are no duplicate labels. A label should be placed below a caption or after a heading (fx after a \chapter).

3.1 Graphs and charts

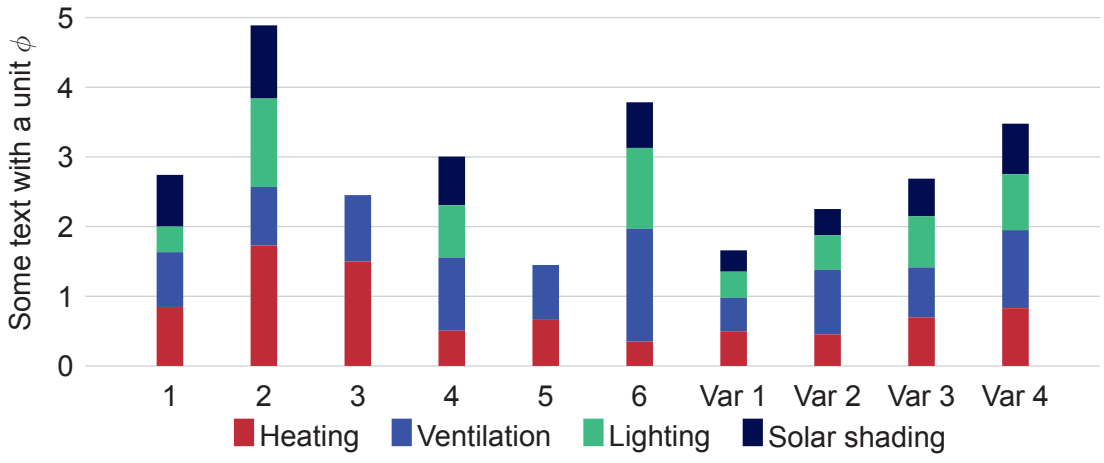

Figure 3.1: Stacked column chart

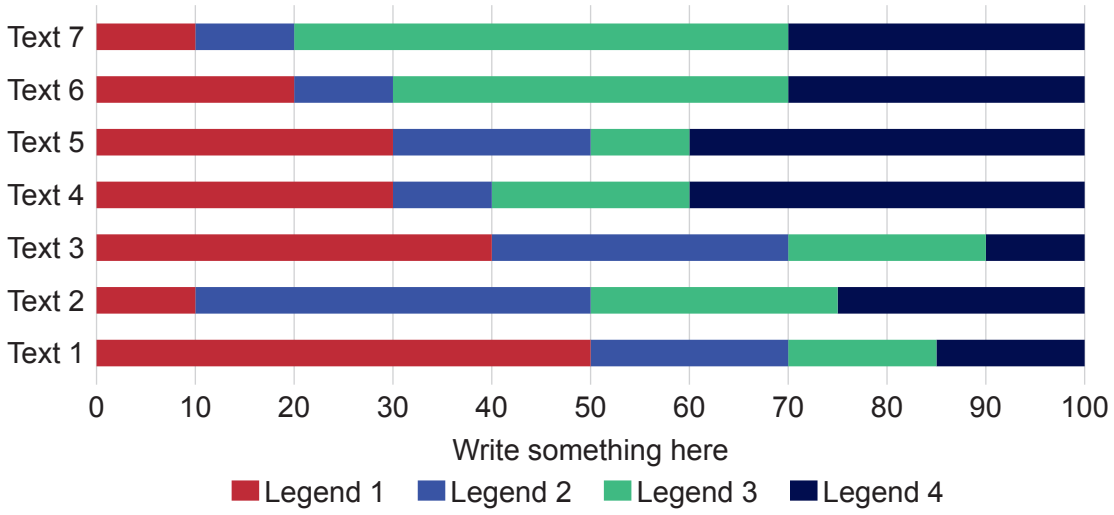

Figure 3.2: Stacked bar chart

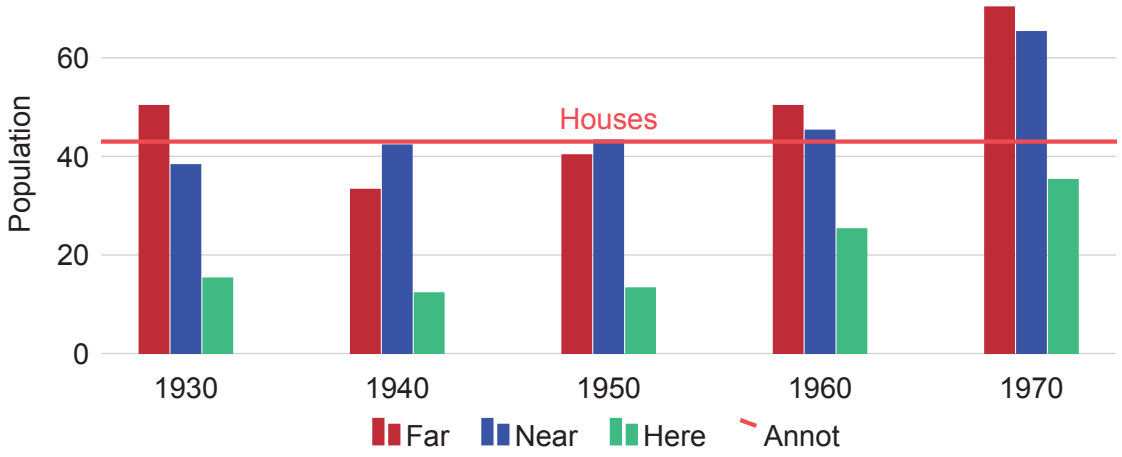

Figure 3.3: Grouped column chart

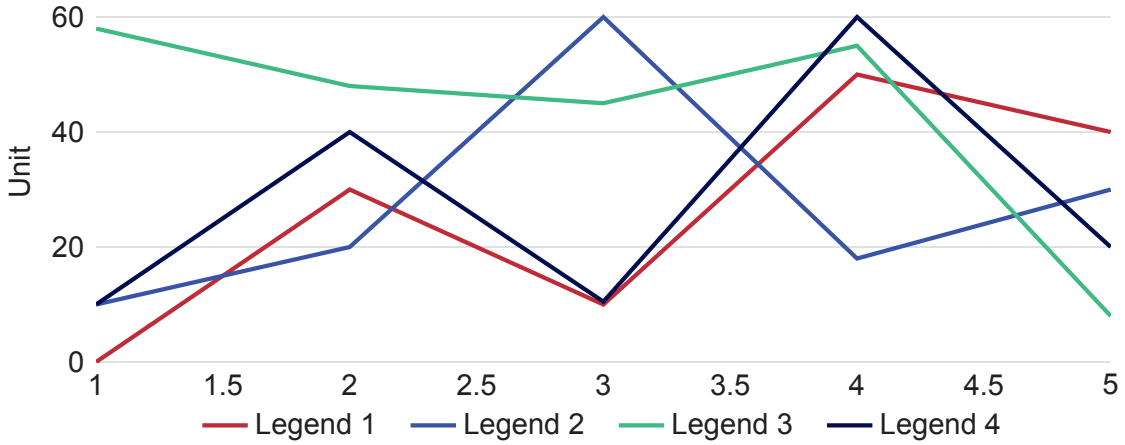

Figure 3.4: Line graph

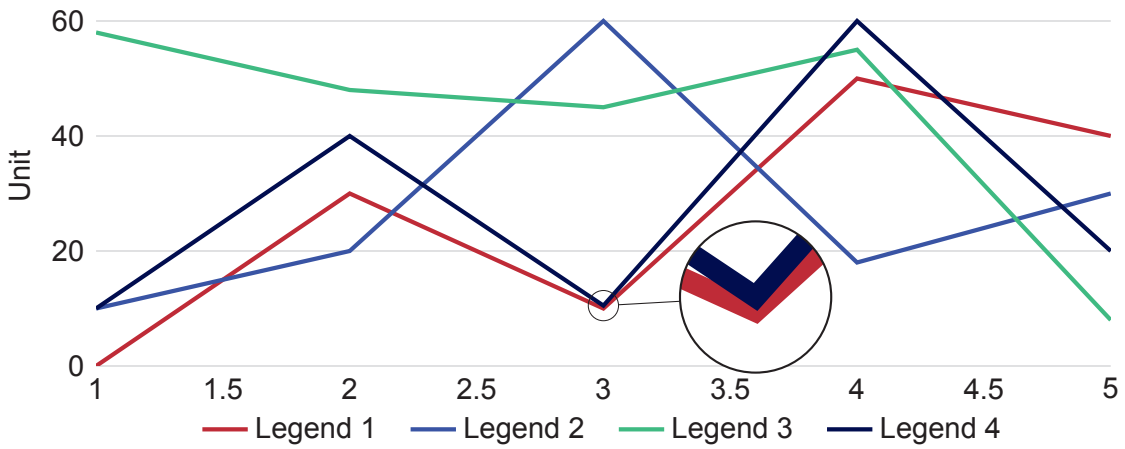

Figure 3.5: Line graph with magnifying glass

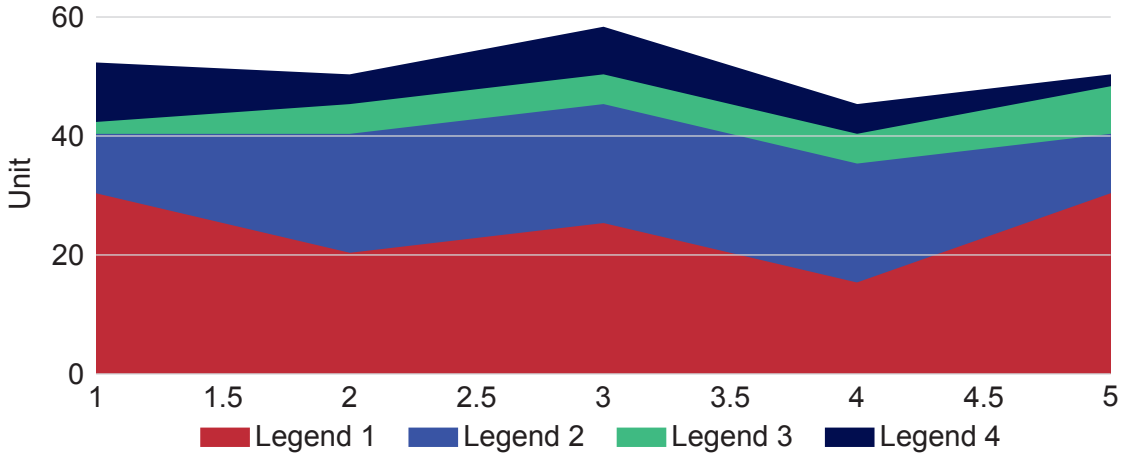

Figure 3.6: Area graph

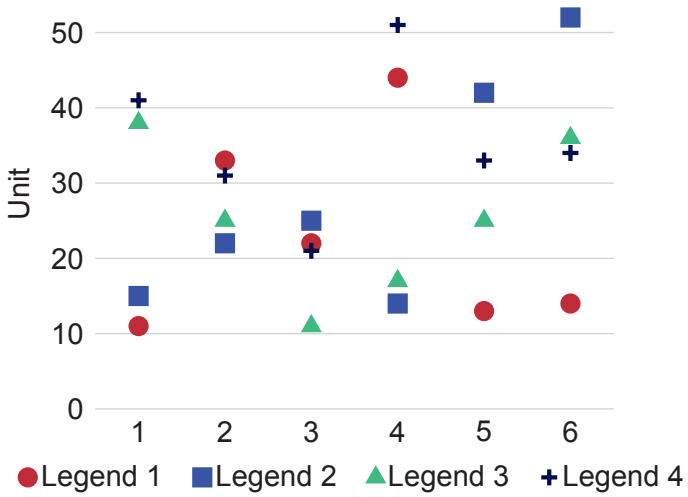

Figure 3.7: Scatter plot

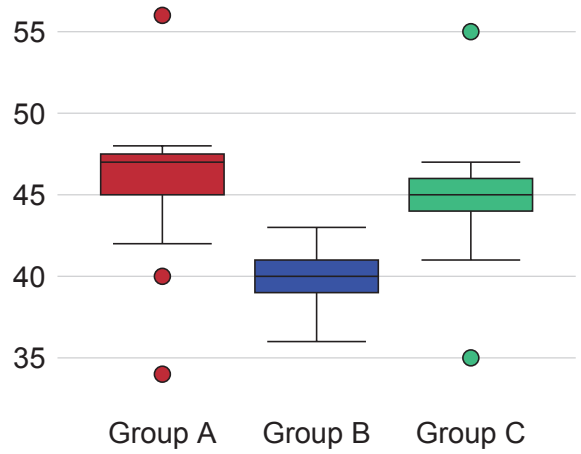

Figure 3.8: Boxplot

3.2 Tables and figures

Table 3.1: This is a `booktabs` table. Go to `http://www.tablesgenerator.com/` and use the booktabs table style

	Item	
Animal	Description	Price($)
Gnat	per gram	13.65
	each	0.01
Gnu	stuffed	92.50
Emu	stuffed	33.33
Armadillo	frozen	8.99

`Booktabs` tables don't use any vertical lines. Only horizontal lines are used. Table 3.1 begins with a `\toprule`, ends with a `\bottomrule` with `\midrule` in between. The table has 3 columns formatted as `@{}llS@{}`. `@{}` is cropping the horizontal lines of the table to fit the content (removes column spacing at the left and right edges). `l` aligns the column to the left and `S` aligns the column according to the decimal point (`siunitx` package). You can of course also use `r` to align right or `c` to center the contents of the column.

Table 3.2: Wrongly formatted table

	Voltage V	Current A	Power W
Transformer input	234.4	0.50	117.4
Transformer output	25.86	2.72	70.3
Efficiency			60%

Table 3.3: Correctly formatted table

	Voltage V	Current A	Power W
Transformer input	234.4	0.50	117.4
Transformer output	25.86	2.72	70.3
Efficiency			60 %

Table 3.2 and table 3.3 have the same comtents but there are some subtle differences in formatting which makes table 3.3 the superior table of the two. The most obvious change is removing the midrule between the transformer input and output rows. The efficency row is the odd man out and a midrule has been used to emphasise the difference between the transformer rows and the efficiency row. The delimiters in the voltage, current and power columns are aligned. The horizontal lines (rules) fits to the content and instead of protruding. The spacing between 60 and the percentage sign is correctly adjusted.

Figure 3.9: Just a normal figure

(a) A subfigure (b) A subfigure

Figure 3.10: A figure with two subfigures

(a) A subfigure

(b) A subfigure

(c) A subfigure

(d) A subfigure

Figure 3.11: A figure with four subfigures

Referring to the figure as a whole fig. 3.11 or to an individual sub figure fig. 3.11a is done the normal way with `\cref{}` commands.

3.3 Equations

In-line math is easy. Anything surrounded by dollar signs becomes a math field. Here is an example: $f(x) = 2x - 1$. Also anything inside the "\begin{equation}" and "\end{equation}" environment is also a math field. Examples are shown below.

All equations use the default latex font. Some might say it looks weird with a serif font for equations and a sans-serif font for the body text. However, it is very unpractical to change the math font in latex which is the exactly the reason why this has not been done. One benefit of the serif style math font is the clear distinction between symbols (variables) and units.

On the subject of units, those are all taken care of by the `\siunitx` package. Whenever there is a number followed by a unit one should write \SI{number}{unit}. Note this command is case sensitive. If a unit should follow a variable use the command \si{unit} (also case sensitive).

The ideal gas law is shown in eq. (3.1).

$$p \cdot V = n \cdot R \cdot T \tag{3.1}$$

$$\frac{\partial}{\partial t} \int_0^\delta U dy = -\delta \frac{1}{\rho} \frac{\partial P}{\partial x} - U_f(t)^2 \tag{3.2}$$

$$d_{step} = \sqrt{\frac{\delta}{\frac{dw}{dp_v}} \cdot t} = \sqrt{\frac{1.0 \times 10^{-11}\,\mathrm{kg/(m\,s\,Pa)}}{\frac{5.4\,\mathrm{kg/m^3}}{233.82\,\mathrm{Pa}}} \cdot 7200\,\mathrm{s}} = 0.001\,766\,\mathrm{m} = 1.766\,\mathrm{mm} \tag{3.3}$$

$$x = \mathtt{x}, \mathbf{x}, \boldsymbol{x}, x_{1_{2_{3_4}}}^{1^{2^{3^4}}} \cdot hello * \text{hello world} \cdot \text{equation without number}$$

Notice how the `aligned` environment can be used to align the equilibrium arrows in eq. (3.4). Only one equation number is generated using this method. Alternatively if you want an equation number for each line see eqs. (3.5) to (3.6).

$$\begin{aligned} CH_3COOH + OH^- &\rightleftharpoons CH_3COO^- + H_2O \\ H_2O &\rightleftharpoons H^+_{(aq)} + OH^-_{(aq)} \end{aligned} \tag{3.4}$$

$$f(x) = 1 + x - 3x^2 \tag{3.5}$$

$$g(x) + y = 3x - \frac{1}{2}x^3 \tag{3.6}$$

3.4 Listings (code)

Listing 3.1 is a nicely formatted block of code. A listing will automatically continue on the next page if it encounters a page break. Many different programming languages can be highlighted. Check the `listings` package documentation for a list of supported programming languages.

```
%% Monte Carlo simulation, estimation of pi
m=1E7;

x=rand(m,1);
y=rand(m,1);

g = x.^2+y.^2-1;

%dots outside
Pf = sum((g)<=0)/m

pi = 4*Pf
```

Listing 3.1: Monte Carlo simulation to estimate the value of π

Bibliography

[1] Philipp Lehman et al. *Biblatex – Sophisticated Bibliographies in LaTeX*. 2018. URL: https://www.ctan.org/pkg/biblatex.

[2] Ian Goodfellow et al. “Generative adversarial networks”. In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[3] Jonathan Ho, Ajay Jain, and Pieter Abbeel. “Denoising diffusion probabilistic models”. In: *Advances in Neural Information Processing Systems* 33 (2020), pp. 6840–6851.

[4] Jascha Sohl-Dickstein et al. “Deep unsupervised learning using nonequilibrium thermodynamics”. In: *International Conference on Machine Learning*. PMLR. 2015, pp. 2256–2265.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-net: Convolutional networks for biomedical image segmentation”. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer. 2015, pp. 234–241.

[6] Christoph Schuhmann et al. “Laion-5b: An open large-scale dataset for training next generation image-text models”. In: *arXiv preprint arXiv:2210.08402* (2022).

[7] Robin Rombach and Patrick Esser. *Hugging Face - Stable Diffusion*. 2023. URL: https://huggingface.co/CompVis/stable-diffusion (visited on 03/02/2023).

A Title

Hello, here is some text without a meaning. This text should show what a printed text will look like at this place. If you read this text, you will get no information. Really? Is there no information? Is there a difference between this text and some nonsense like “Huardest gefburn”? Kjift – not at all! A blind text like this gives you information about the selected font, how the letters are written and an impression of the look. This text should contain all letters of the alphabet and it should be written in of the original language. There is no need for special content, but the length of words should match the language.

Technical
University of
Denmark

Brovej, Building 118
2800 Kgs. Lyngby
Tlf. 4525 1700

www.byg.dtu.dk