

---

# SpaceSheets: Interactive Latent Space Exploration through a Spreadsheet Interface

---

Bryan Loh and Tom White

School of Design

University of Wellington

Wellington, New Zealand

{lohbrya@myvuw.ac.nz, tom.white@vuw.ac.nz}

## Abstract

Generative models capture properties and relationships of images in a generic vector space representation called a latent space. Latent spaces can be sampled to create novel images and perform semantic operations consistent with the principles inferred from the training set. Designers can use representations learned by generative models to express design intent enabling more effective design experimentation. We present the SpaceSheet, a general-purpose spreadsheet interface designed to support the experimentation and exploration of latent spaces.

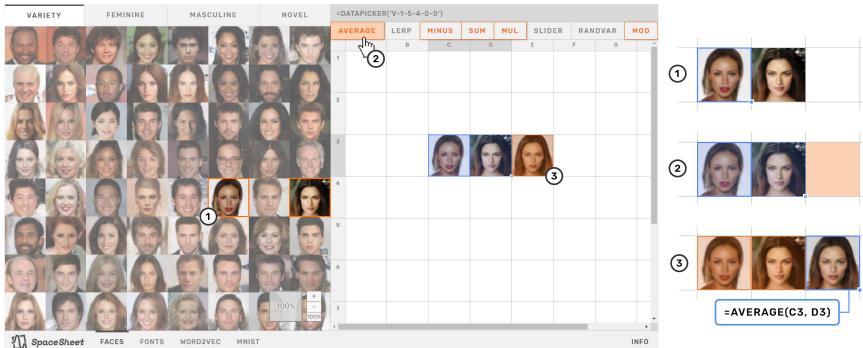


Figure 1: The SpaceSheet being used to perform an average between two latent variables.

## 1 Introduction

Problem solving can be viewed as a search for a solution within a space. In design, this process involves generating solutions and evaluating their consequences relative to goals and constraints (Simon, 1995).

Generative models learn more compact representations of the training data in a vector space of latent variables. Latent variables are sampled from high-dimensional latent space and can be decoded back into observable values. Additionally, semantic operations can be performed within latent space using vector arithmetic (White, 2016).

Spreadsheet interfaces are a ubiquitous part of office productivity suites. They enable users to perform experimental calculations using a set of formulae which define relationships spatially. Automatic recalculation supports experimentation by enabling users to observe the results of their actions immediately and act accordingly.

We developed SpaceSheet (Figure 1) to leverage the familiarity and power of spreadsheet interfaces for the purpose of design experimentation within latent space. It has been adapted to enable non-experts to explore and experiment within latent spaces.

## 2 SpaceSheet

SpaceSheet consists of a data picker exposing latent variables to operate with and a spreadsheet to define operations between the variables. In both, latent variables are decoded into observable images.

### 2.1 Data Picker

The data picker is a predetermined set of latent variables which have been organized into a grid. The set of variables in the data picker act as the points of reference from which the latent space can be explored from. Diversity has been prioritized in the selected set to maximize the variety of possible outcomes that can be explored. Multiple data pickers have also been implemented as tabs to provide various pre-baked distributions of latent variables.

### 2.2 Spreadsheet

The spreadsheet is the main workspace of the tool. It enables users to express relationships between cells using formulae. Operations between cells containing latent variables are computed with vector arithmetic, and its result is decoded into an image. Common operations can be defined by clicking on buttons at the top of the spreadsheet. These buttons are selection-aware, and highlight to suggest operations based on the selected cells. A live SpaceSheets demo is available online<sup>1</sup> and the appendix contains a list of supported operations and sample workflows.

## 3 Applications



Figure 2: Fonts explored with interpolation (left). Computing an analogy with Word2Vec (right)

Initial efforts are focused on experimenting in various domains to encourage the development of a general-purpose model agnostic set of operations. A SpaceSheet to explore a generative model of fonts (Bernhardsson, 2015) has been implemented to be used as a design tool (Figure 2). User testing indicated that the tool enabled designers and non-designers alike to explore design variations easily (Loh 2018).

The concepts have also been extended to domains other than images and with models that are not generative, such as the Word2Vec model (Mikolov, Chen, Corrado, and Dean, 2013). This version of the SpaceSheet can be used to find word analogies and perform interpolations using nearest neighbors (Figure 2).

We are continuing to enhance the capabilities of the tool by extending the family of neural models that can be explored and supporting the various use cases that arise.

<sup>1</sup><https://vusd.github.io/spacesheet/>

## References

- Bernhardsson, E. (2018). deep-fonts: Generate fonts using deep learning. Python. Retrieved from <https://github.com/erikbern/deep-fonts> (Original work published 2015)
- Liu, A. (2018). Word2Vec JSON. JavaScript. Retrieved from <https://github.com/turbomaze/word2vecjson> (Original work published 2016)
- Loh, B. (2018). SpaceSheets: Design Experimentation in Latent Space. Victoria University of Wellington, Wellington, New Zealand.
- Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv:1301.3781 [Cs]. Retrieved from <http://arxiv.org/abs/1301.3781>
- Simon, H. (1995). Problem Forming, Problem Finding, and Problem Solving in Design. *Design and Systems; General Applications of Methodology*, 3(828), 245–257.
- Tian, Y. (2018). GAN Showcase. Python. Retrieved from <https://github.com/alantian/ganshowcase> (Original work published 2018)
- White, T. (2016). Sampling Generative Networks. arXiv:1609.04468 [Cs, Stat]. Retrieved from <http://arxiv.org/abs/1609.04468>

## Appendix: Implementation Details

### Supported Operations

Operation	Description	Formula
Sum	Adds a list of numbers / variables	SUM(val1, val2, val3, ...)
Minus	Subtracts two numbers / variables in sequence	MINUS(val1, val2)
Multiply	Multiplies a list of numbers / variables	MUL(val1, val2, val3, ...)
Linear Interpolation	Calculates the value in between two numbers / vectors at a specified amount	LERP(from, to, amount)
Average	Calculates the average of a list of numbers / vectors	AVERAGE(val1, val2, val3, ...)
Distance	Calculates the euclidean distance between two numbers / vectors	DIST(val1, val2)
Modulate	Creates a scrubbing interface which can modulate a cell	MOD(cell, degree, radius)
Random Variable	Creates a random latent variable	RANDVAR(seed)
Slider	Creates a number which is controlled by a slider element	SLIDER(min, max[, step])

## Interactive Cell Types

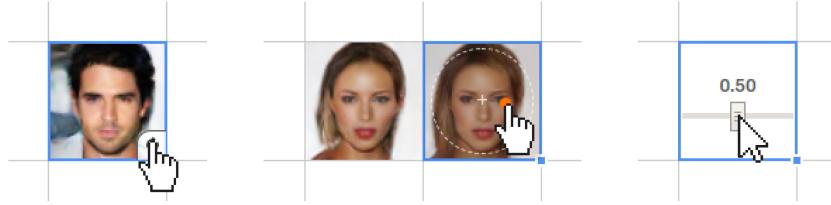


Figure 3: RANDVAR, MOD and SLIDER cells.

Several alternative cell types have been implemented to create interface elements which support more effective search and exploration. These are instantiated by the operations:

`RANDVAR(seed)`

The RANDVAR (random variable) cell instantiates a latent variable from a random seed. This enables users to operate using latent variables beyond the limited set afforded by the Data Picker. A button displays when the cell is hovered over which enables users to randomise the cell directly.

`MOD(base, degree, distance)`

The MOD (modulate) cell exposes a joystick interface which enables users to scrub locally around a given latent variable to arrive at similar latent variables. The degree of difference can be controlled by the joystick's distance from the center of the cell.

`SLIDER(min, max [,step])`

The slider cell enables users to create a number controlled by a slider element.

## Example Workflows

### Interpolation



Figure 4: An interpolation between two latent variables

### Interpolation Strip - Slider Control

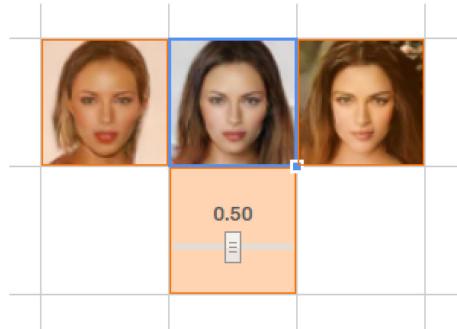


Figure 5: An interpolation between two latent variables controlled using a slider cell

### Interpolation Grid - Four Corners

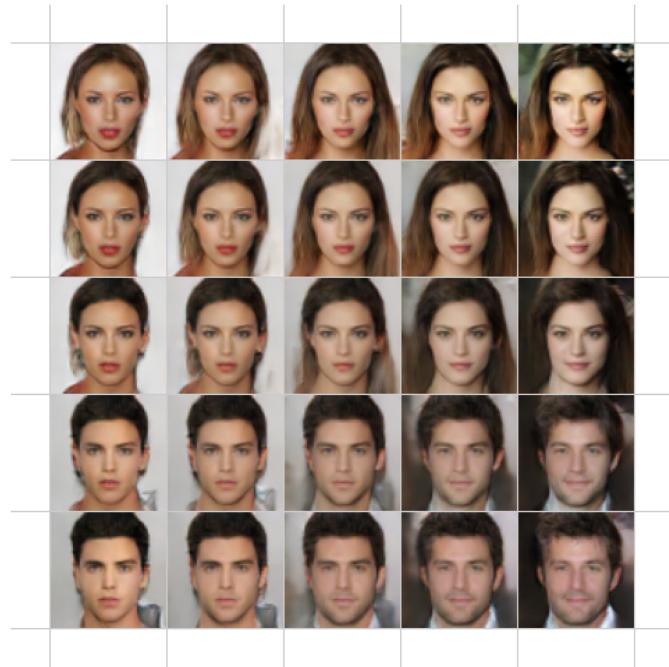


Figure 6: An interpolation between four latent variables

## Analogy

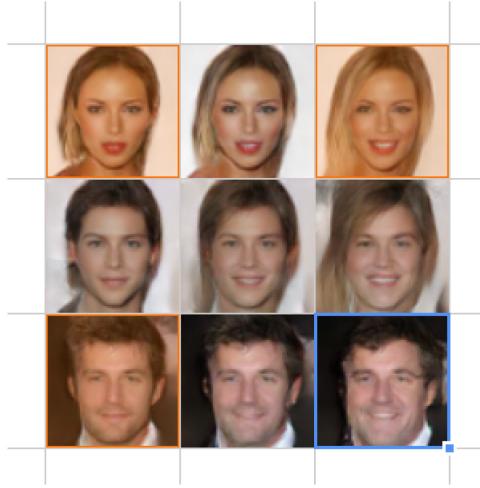


Figure 7: An analogical construction. The bottom right cell applies the difference between the top cells to the cell on the bottom-left

Given three reconstructions (top-left, top-right, bottom-left), the SpaceSheet calculates the bottom-right corner by analogy. This is achieved by applying the difference between the top variables to the bottom-left variable. In this example, a toothy grin has been applied to the man.

## Extrapolation

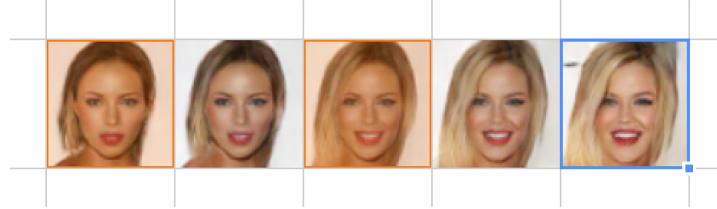


Figure 8: Extrapolating from two points.

Extrapolating from latent variables can be used to emphasise attributes which vary between its anchors. In this example, the difference between the highlighted anchors - blond hair, large smile, etc. - have been emphasised by extrapolating beyond the end anchor.

## Attribute Vectors

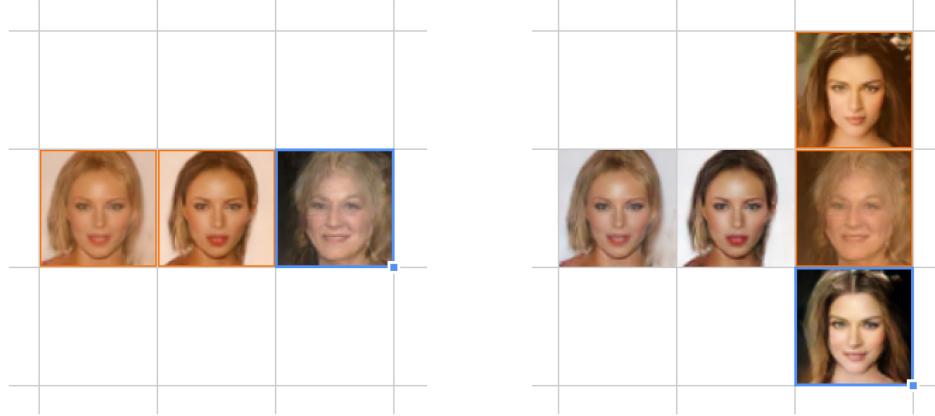


Figure 9: Isolating a ‘blonde’ vector by subtraction (left). Adding the attribute vector to a new latent variable (right)

Specific attributes can be applied as operations to latent variables. Attribute vectors can be isolated by subtracting a latent vector with desired attributes with one without the attributes. This attribute vector can be added to another latent variable to apply the isolated attribute. The example image shows this two-step process. In the first, a ‘blonde’ attribute vector has been isolated by computing the difference between the highlighted cells. This vector is then applied in the right image by addition. The result is a more blonde version of the initial latent variable.

## Averaging

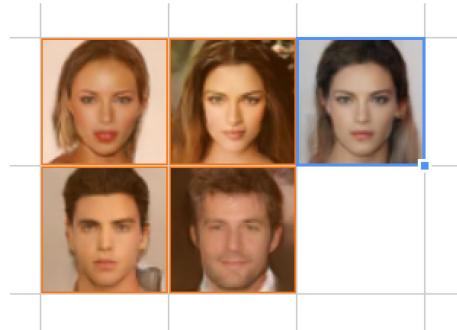


Figure 10: Calculating the average reconstruction of a group of latent variables

## Random Neighbors

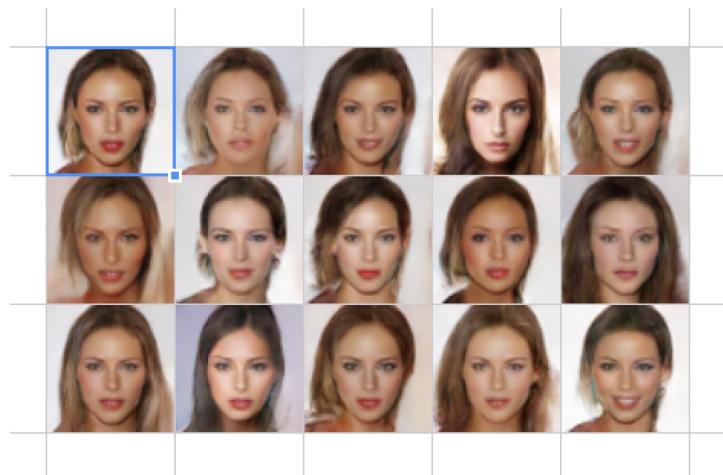


Figure 11: A random sampling around the selected cell

A random sampling of neighbouring variables can be created using the MOD operation. In the image, the selected cell formed the base from which MOD cells are created.