Business & Decision

Deep Learning 1

Raoul Grouls, 25 april 2022

What is intelligence?

Exercise:

- 3 minute break out
- Find a definition of intelligence

What is AI?

What is AI?

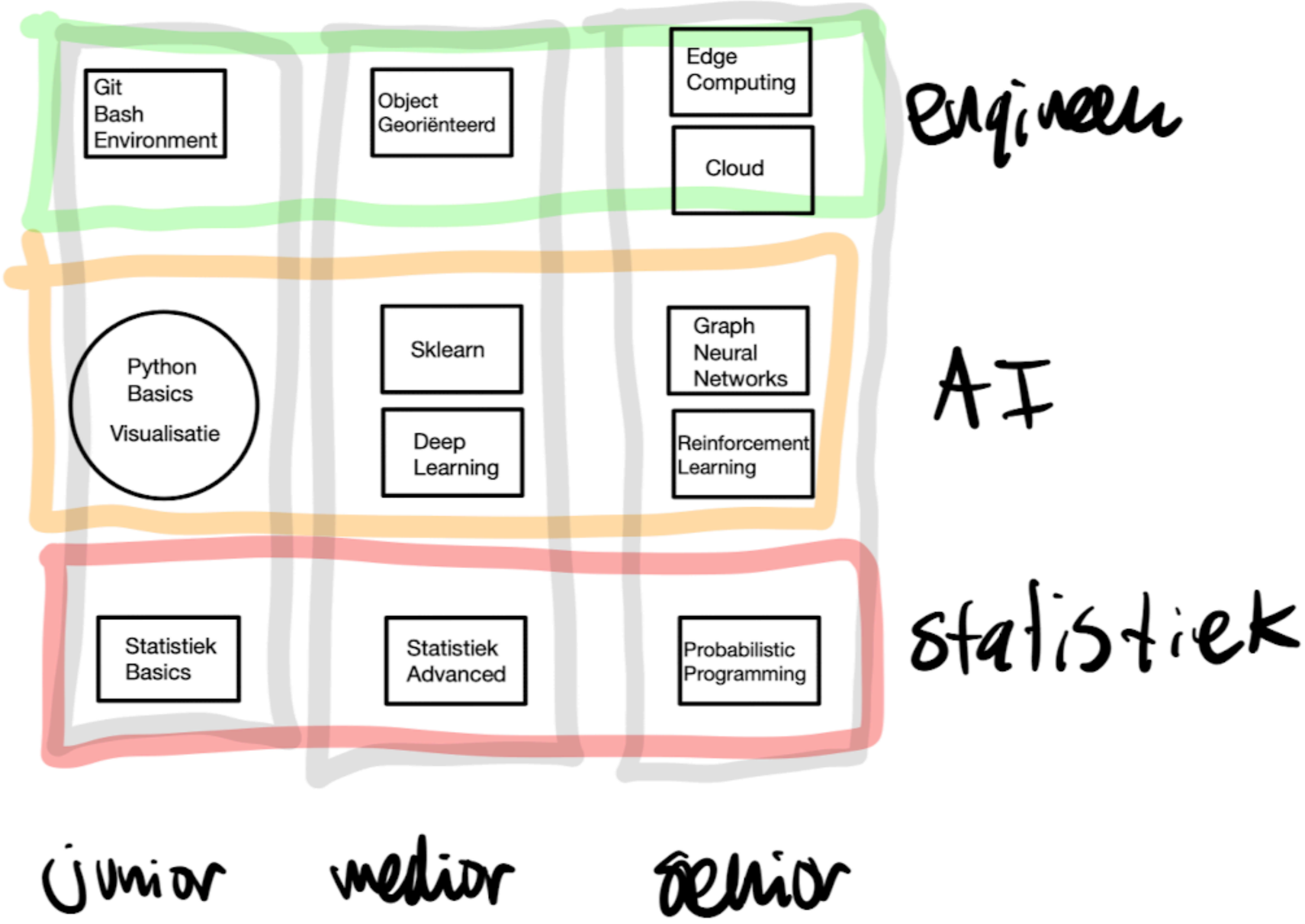
Exercise:

- 3 minute break out
- Identify observations and labels in your work field
- Identify employees that have the job of connecting observations and labels

What is deep learning?

A machine learning technique that uses **multiple layers of non-linear transformations** in order to connect observations and labels

A technique to solve problems by providing observations and labels, letting the computer find the solution using neural networks.


Git
Bash
Environment
Object
Georiënteerd
Edge
Computing
Cloud
Engineer
Python
Basics
Visualisatie
Sklearn
Deep
Learning
Graph
Neural
Networks
Reinforcement
Learning
AI
Statistiek
Basics
Statistiek
Advanced
Probabilistic
Programming
statistiek
Junior
medior
senior

A very short history of AI

- 1950 Mathematics of Neural Networks invented
- 1990-2010 Support Vector Machines, Random Forest (Machine learning)
- 2010 Deep learning
- 2015 Graph Convolutional Networks
- 2017 Attention mechanisms

Complexity 1990 - SVM

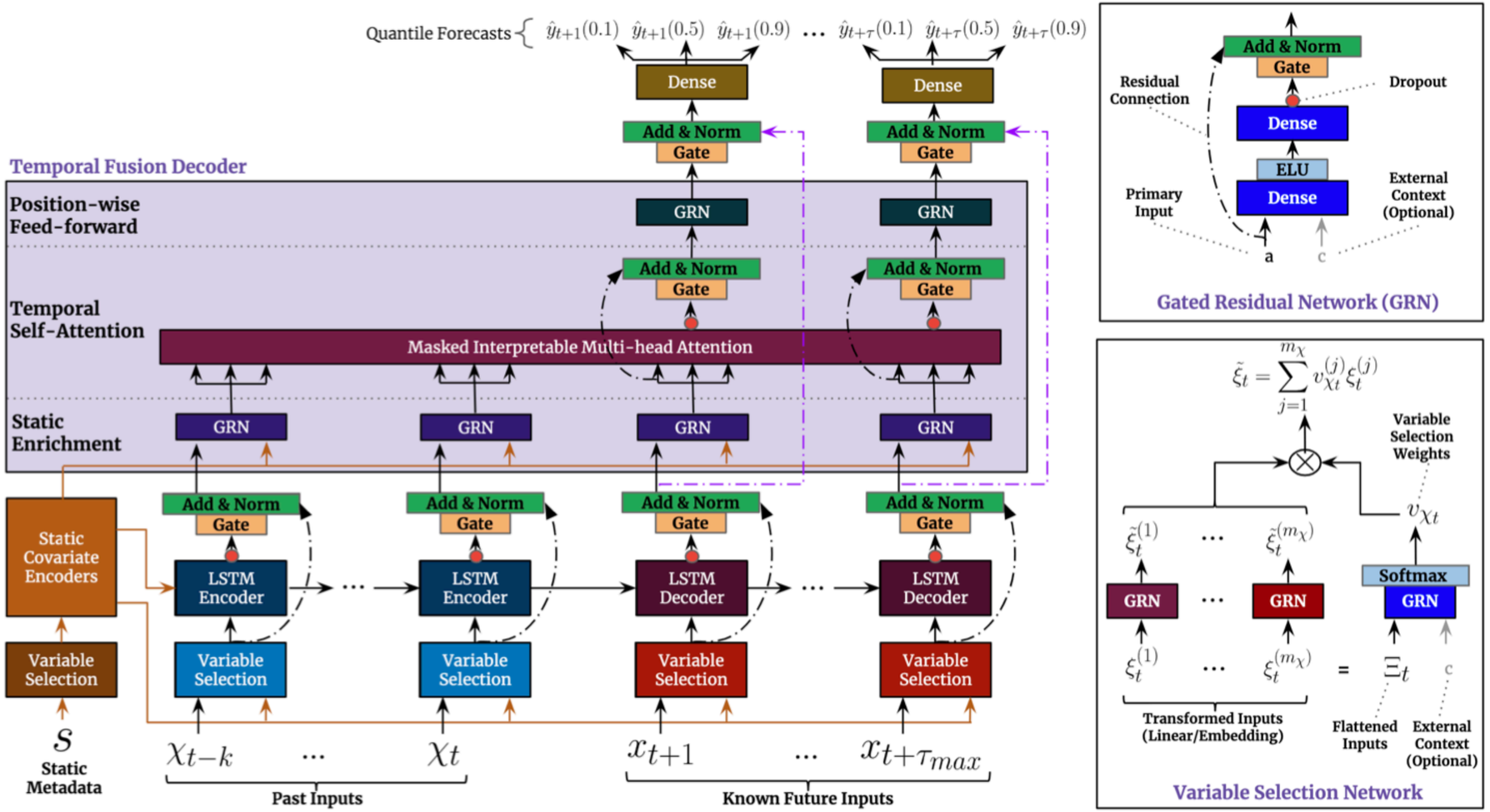
$$max_a \sum_{i=1}^{m} a_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} a_i a_j y_i y_j K(x_i, x_j)$$

$$K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2)$$

$$J(w, b) = \frac{1}{2} w^T w + C \sum_{i=1}^{m} max(0, 1 - t^i(w^T x^i + b))$$

m	number of observations
a	counter for errors
x	observations
y	labels
γ, C	hyperparameters
w, b	weights we need to learn

Complexity 2021 - TFT



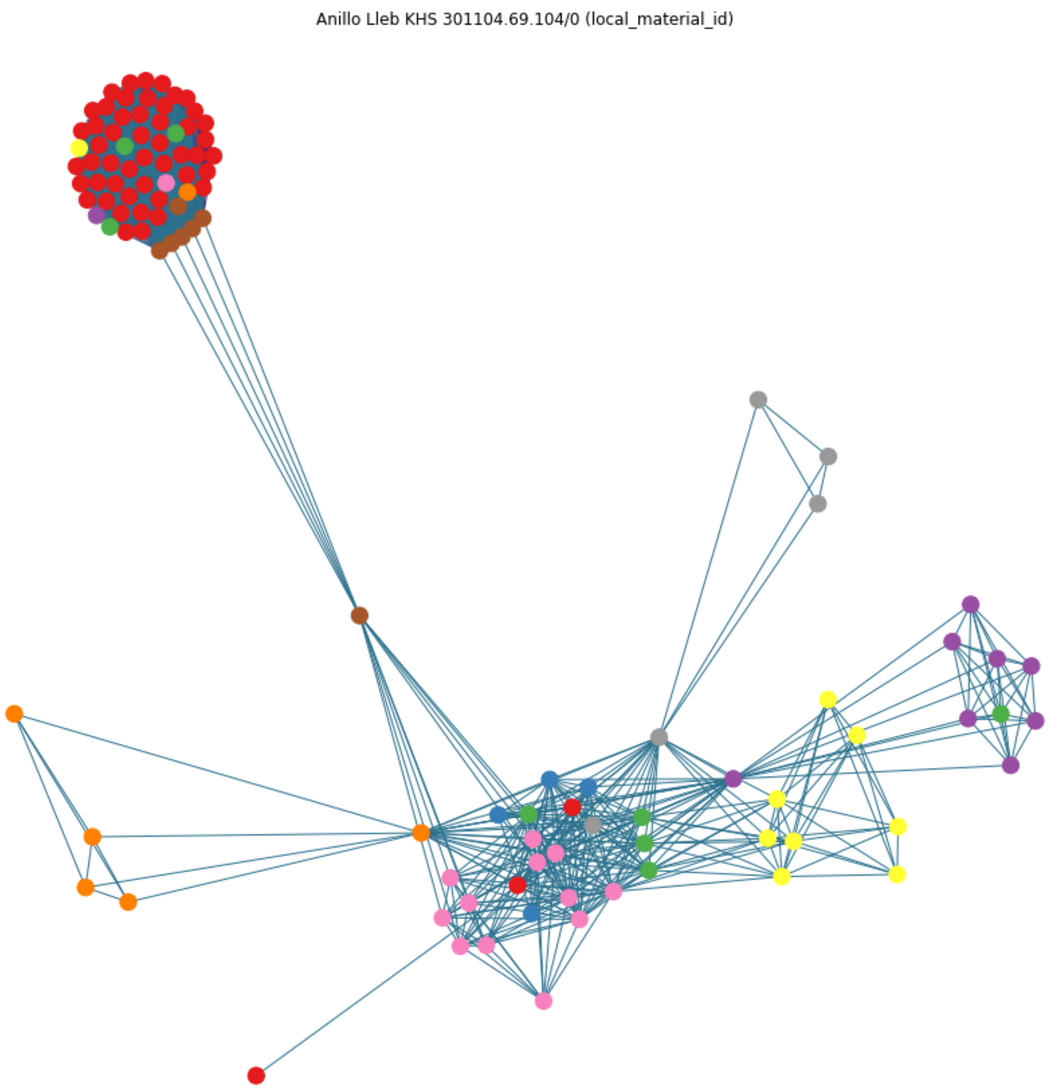
Examples

Graphs in factories

Problem:

Relocate 30.000 items within 6 weeks to European factories.

Solution: Locate similar machine parts in a database of 300.000 items, with unstructured and noisy descriptions in 10+ languages.

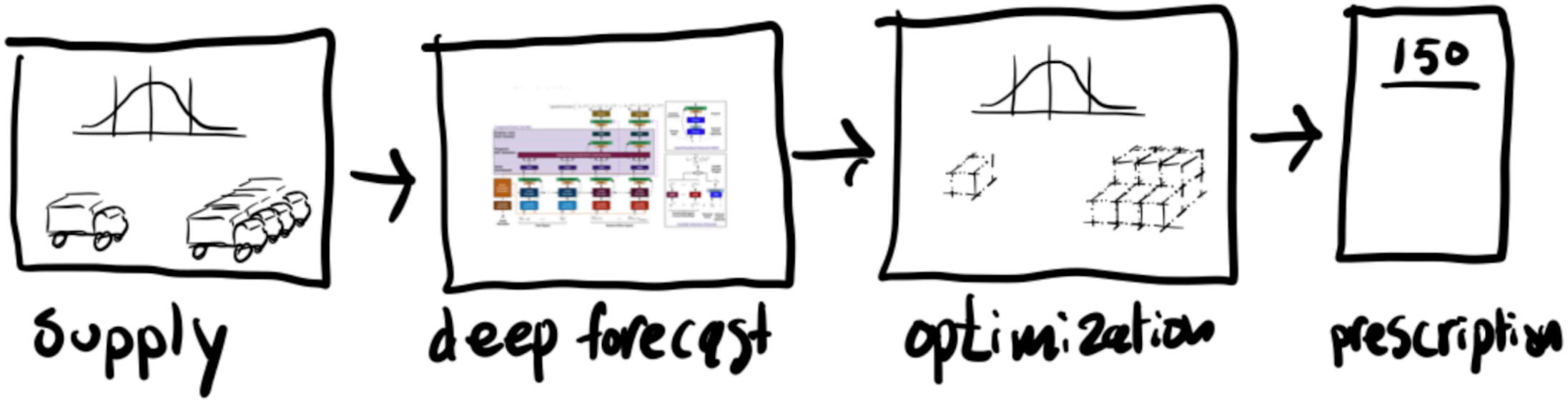


Examples

Probabilistic Stock Optimization

Problem: find optimal stock levels, while supply and demand have a lot of uncertainty.

Solution: build a probabilistic prescriptive model



Main libraries

- Numpy
- Pandas
- Scipy
- Matplotlib / Seaborn
- PyTorch
- Trax/Jax

Code style standards

- Use pathlib.Path instead of os.path
- Use loguru instead of print
- Use pydantic for dataclasses and settings
- Use typehinting and linting
- Organize your code with classes
- Create custom types with Protocol or pydantic
- Manage your environment with poetry, not conda
- Use de data science cookiecutter template

See https://staticsitecodestyle.z6.web.core.windows.net/tldr.html

Linear models

Even simpler models are called “retarded”

Linear models are one of the simplest models available. Their advantages are:

- The simplicity helps us to avoid overfitting
- They are fast and we don’t have to worry about getting stuck in a local minimum.
- They are ideal as a baselinemodel

Linear models

The basic mathematical shape of a linear model is:

$$Y = WX + b$$

- $Y = \{y_1, \ldots, y_m\}$ are labels, so we have m labels.
- Every label Y_i has corresponding features $X_i = \{x_{i,1}, \ldots, x_{i,n}\}$, so we have n features.
- We can store the observations in an (m, n) matrix X
- We can store n weights for every feature in a matrix W
- b is an additional bias weight

The matrix notation is concise. We could write this out in full as:

$$y_i = w_1 x_1 + w_2 x_2 + \ldots + w_n x_n + b$$

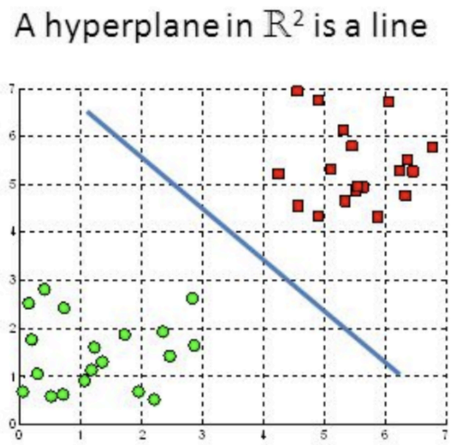
Hyperplanes

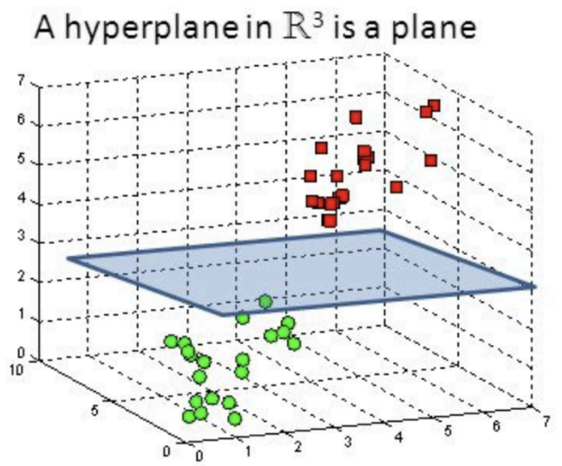
It's a bird… It's a plane…

- For classification, our outcome are discrete classes (e.g. "yes" or "no")
- For regression, our outcome is a real number (e.g. 1.4 or 26.834)

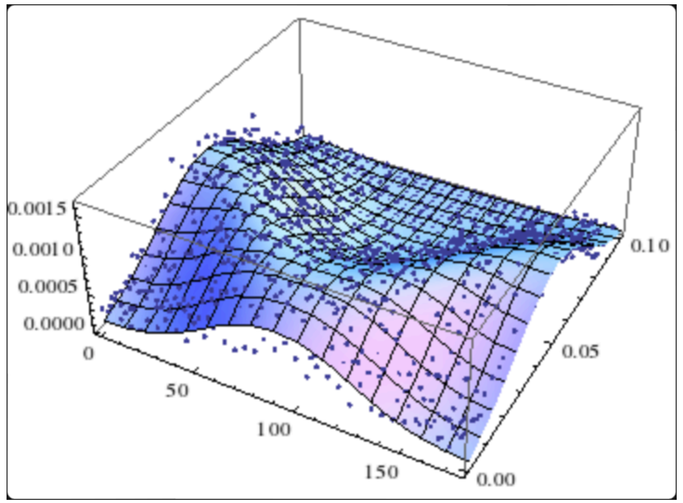
We can use a linear model for both cases. We call these models "hyperplanes": they have one dimension less than the ambient space.

With classification, we want the data to be separated by the hyperplane. With regression, we want to points to get as close as possible to the hyperplane





Hyperplane for classification



Hyperplane for regression

Non-linear models

A lot of data is non-linear. This means we would need a curved hyperplane.

One trick to do this is the kernel-trick, which is commonly used with Support Vector Machines, which we touched upon in the short history.

Deep learning has found another trick, we will look at the labs to see how this works.