OPM 562
Case study:

Supervised learning for data driven tomato yield prediction and control of greenhouses

Son Phan, 1713240

26/05/2020

Contents

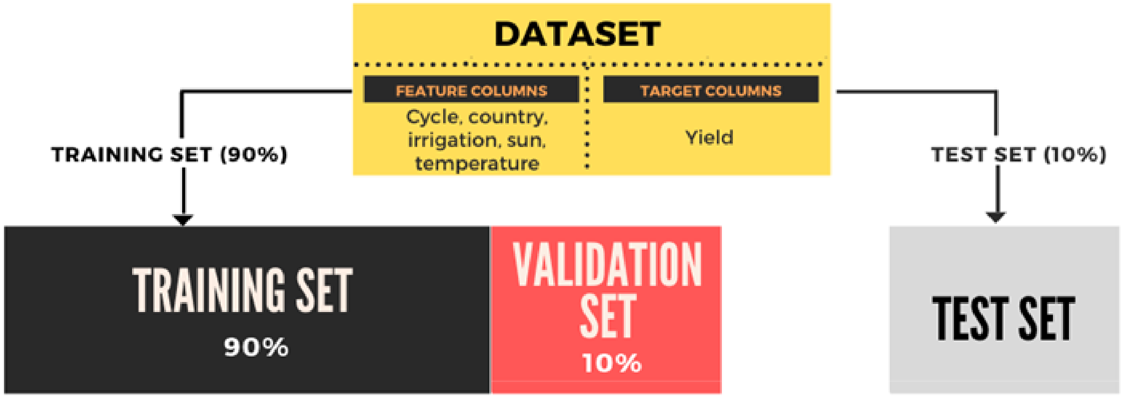
1. Data Preparation

1. Convert categorical data
   a. Cycles (C1-C6) - to dummies
   b. Countries: Spain = 0, Netherlands = 1

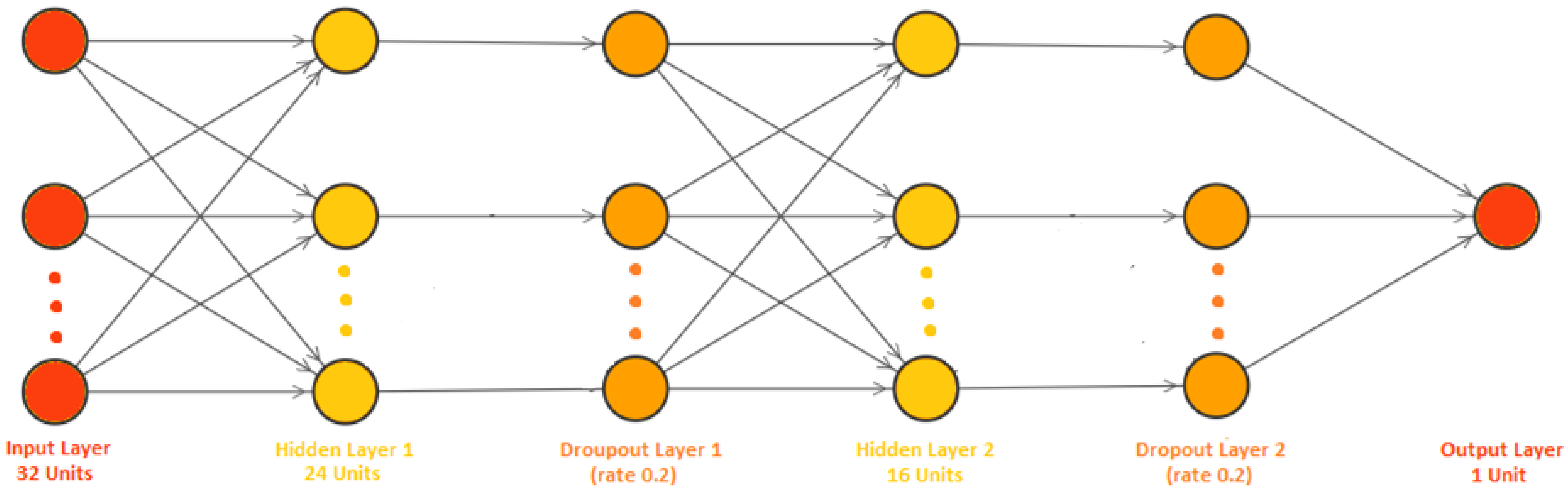
2. Separate feature & target columns

3. Train-validation-test split

Larger amount of training data makes the NN better understand data distribution.



2. Neural Network



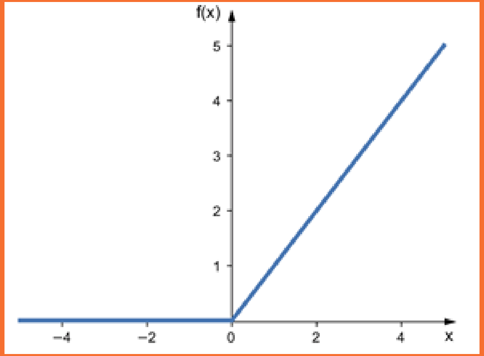
2.a. Architecture and Structure

Feature	Decision	Justification
Architecture	**Feed-forward neural network**	• **Arguments for a FFNN** ○ Most suitable for regression prediction problems where a numerical value (target) is predicted given a set of inputs (features) ○ Data to be learned is neither sequential nor time-dependent.
Structure	**2 hidden layers** **1561 parameters**	• **Layers (Depth)** ○ Non-linearity in the data ○ Lack of generalization for shallow models ○ Test error does not improve anymore after two layers • **Units (Width)** ○ No. of units in hidden layers ≤ no. of units in input layer ○ Units add up to parameters, parameters should approximate total of data points of training set

2.b. Activation Function

Rectifier Linear Unit (ReLU) Function

- Used on every hidden layer
- Most suitable for a regression problem
- Results in a numerical value > 0
- Alleviates the problem of vanishing gradients in deep models



2.b. Loss Function

Mean Squared Error (MSE)

- Most suitable for a regression problem
- The model is punished for making larger mistakes → optimizes accuracy of our prediction
- Preferred loss function as output type is continuous numerical value and distribution of target variable is Gaussian
- Keep track of MAPE to check the network performance

2.c. Prevention of overfitting

L2 regularization

- L2 regularization smooths the parameter distribution
- High performance when combined with dropout regularization (Srivastava et al. 2014).

Why only L2 and not L1?

- Less computationally expensive
- Avoid feature selection

Dropout layers

- Large neural nets trained on relatively small datasets can overfit the training data.
- Dropout layer:
  - Simulates training a large number of neural networks
  - Makes training process noisy
  - Increases generalization power
- Dropout layer assigned to each hidden layer
- Common rate range cited in literature: [0.2 - 0.5]. We decided to use 0.2 because of the size of our data.

2.d. Training Hyperparameters

Batch size

Mini-batch gradient descent

- Split training set into smaller sets
- Implement gradient descent on each batch one after the other
- Mini-batch size should be smaller than number of datapoints in training data
- Increased size to compensate for high number of epochs

Faster & more efficient algorithm

Epochs

- Increase to compensate for the “noises” that dropout layers add to training process
- Train as long as validation error decreases

Avoids over- or underfitting

Learning rate

- Adam maintains and adapts learning rates for each of the weights in the model
- Computationally efficient, little memory requirement

Low training cost over iterations (compared to other optimizers)

2.e. Training results

MSE/MAPE in training & validation data over epochs

After 10,000 Epochs:

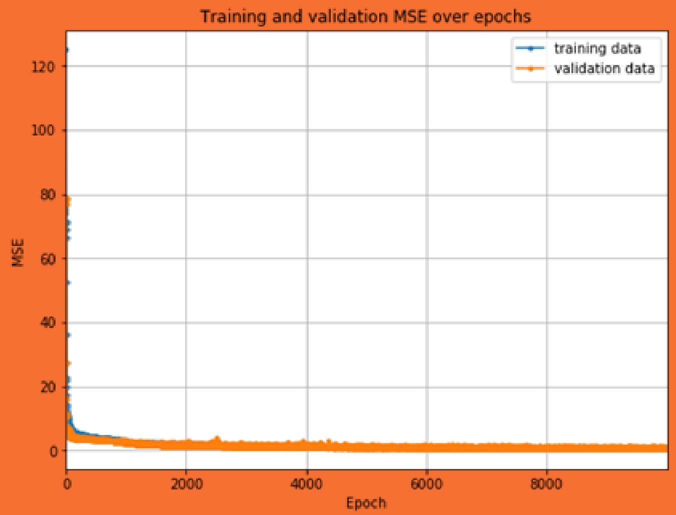
- ➔ Validation error shows no further decrease
- ➔ Training error and validation error converge (small generalization gap)
- ➔ Stop early before overfitting

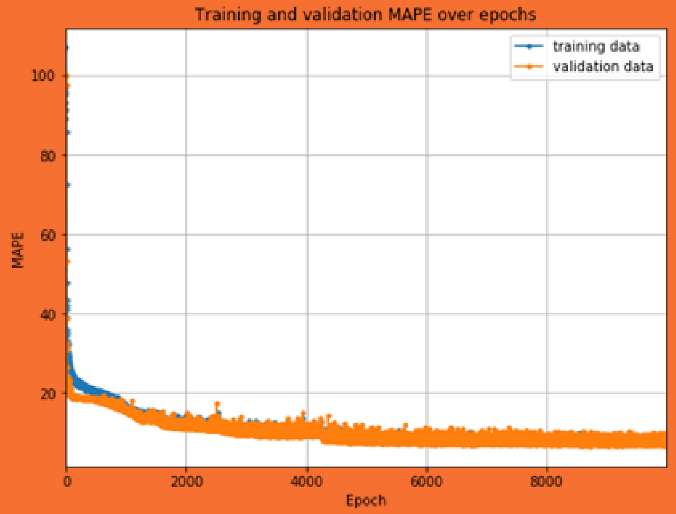
Model evaluation

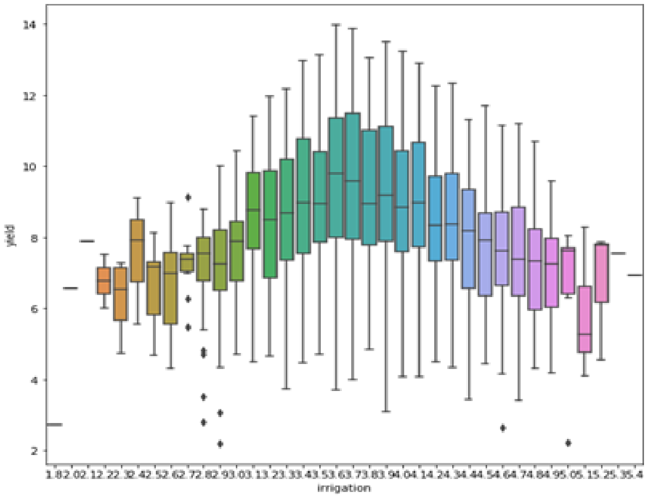
	Training	Validation	Test
MSE	0.62	0.79	0.75
MAPE	6.90 %	8.00 %	6.99 %

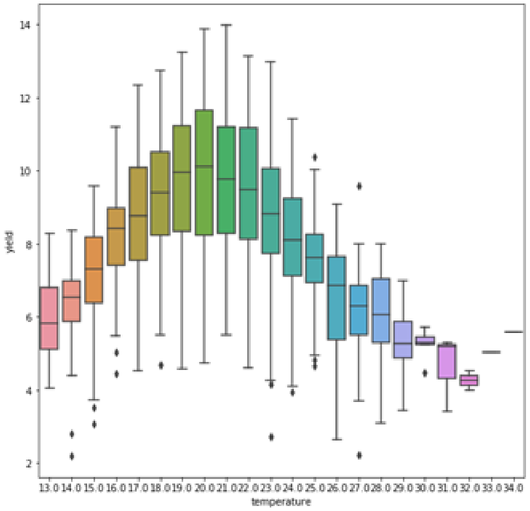
➔ Network performs well on test data: MAPE below 10 % indicates very high forecasting accuracy

Source: Lewis 1982, p.40









3.a. Parameter configurations

Ranges for the parameters "daily irrigation" and "temperature inside" are taken from min and max values of these parameters in the provided dataset.

	min value	max value	step
area	1000 m^2	50000 m^2	1000 m^2
pesticide	0	1	1
daily irr.	1.5 L/m^{2}d	5.5 L/m^{2}d	0.5 L/m^{2}d
t° inside	13 °C	34 °C	1 °C

19800 different parameter configurations

3.b. Cost function

Total cost = irrigation + penalty + conditioning + greenhouse costs

0.000021 x area x daily irrigation x 60

*1 x (demand - area x predicted yield)**
*0***

3600 x | tº inside - tº outside |

20 x area ÷ 6

** if production < demand* ***if production ≥ demand*

3.c. Recommendations
Cycle 2

best	31 modules	Envidum	18 (+3) °C	4.0 L/m²d	✗	-1,806 kg	Backlog
alternative	32 modules	Envidum	18 (+3) °C	4.0 L/m²d	✓	7,814 kg	Leftover
best	€ 103,333	–	€ 10,800	€ 156	€ 1,806	€ 116,095	Total cost
alternative	€ 106,667	–	€ 10,800	€ 161	0	€ 117,628	

3.c. Recommendations
Cycle 3

49 modules	Envidum	21 (-1) ºC	3.5 L/m^{2}d	✓
€ 163,333	–	€ 3,600	€ 216	0

Total cost
€ 167,149

Leftover stock
2,874 kg

3.c. Recommendations
Cycle 4

50 modules	Envidum	22 (-3) °C	4.0 L/m²d	
€ 166,667	–	€ 10,800	€ 252	€ 16,328

Total cost
€ 194,047

Backlog
-16,328 kg

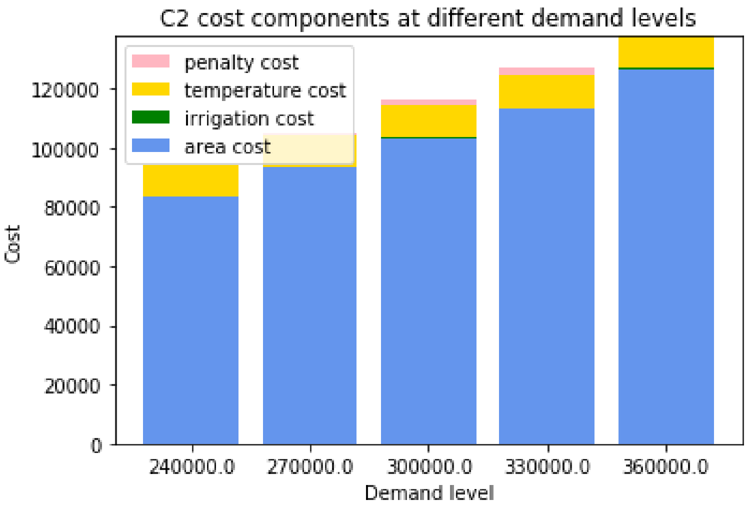
Relatively little sun in Germany

Impossible to meet demand within the given constraints with any costs

Don't accept so large orders!

Expand capacity

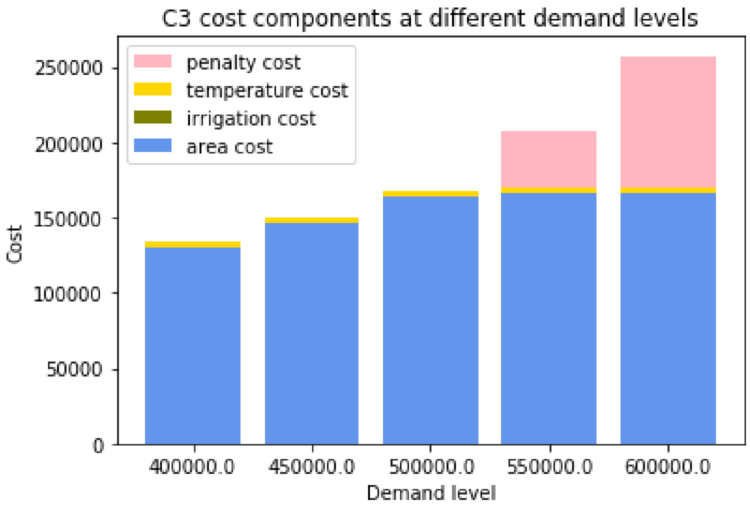
4. Sensitivity analysis
Cycle 2



			Total cost
-20%	25 modules	✓	€ 94,259
-10%	28 modules	✗	€ 104,938
	31 modules	✗	€ 116,095
+10%	34 modules	✗	€ 127,253
+20%	38 modules	✓	€ 137,658

No change: pesticide, irrigation, temperature

4. Sensitivity analysis
Cycle 3

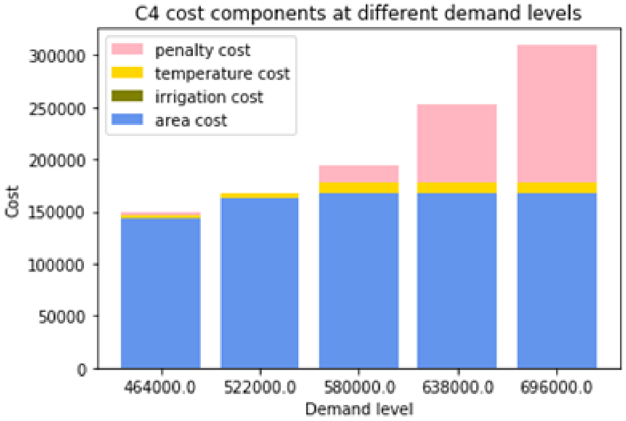


			Total cost
-20%	39 modules	✓	€ 133,772
-10%	44 modules	✓	€ 150,461
	49 modules	✓	€ 167,149
+10%	50 modules	✗	€ 207,351
+20%	50 modules	✗	€ 257,351

No change: pesticide, irrigation, temperature

4. Sensitivity analysis Cycle 4

					Total cost
-20%	43 modules	24 (-1) °C	3.5 L/m^{2}d	✗	€ 148,763
-10%	49 modules	24 (-1) °C	3.5 L/m^{2}d	✓	€ 167,149
	50 modules	22 (-3) °C	4.0 L/m^{2}d	✗	€ 194,047
+10%	50 modules	22 (-3) °C	4.0 L/m^{2}d	✗	€ 252,047
+20%	50 modules	22 (-3) °C	4.0 L/m^{2}d	✗	€ 310,047



No change: pesticide


THANK YOU