MACHINE LEARNING IN SYSTEMS BIOLOGY I (UNSUPERVISED)

Jonas Pleyer

29.06.2022

Freiburg Center for Data Analysis and Modeling (FDM)

INTRODUCTION

’A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E’
- Tom M. Mitchell [Mit97]

1943 First publication of neural network [MP43]

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1966 Following years: Many setbacks in Artificial Intelligence called 'AI-Winters'

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1966 Following years: Many setbacks in Artificial Intelligence called 'AI-Winters'

1995 Support Vector Machines are first introduced

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1966 Following years: Many setbacks in Artificial Intelligence called 'AI-Winters'

1995 Support Vector Machines are first introduced

2002 Torch first release (open source library)

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1966 Following years: Many setbacks in Artificial Intelligence called ’AI-Winters’

1995 Support Vector Machines are first introduced

2002 Torch first release (open source library)

2006 Geoffrey Hinton coins ’Deep Learning’ [HOT06]

1943 First publication of neural network [MP43]

1956 Dartmouth Summer Research Project (Birthplace of modern Machine Learning)

1965 Nilson Machine Learning for pattern classification [Nil65]

1966 Following years: Many setbacks in Artificial Intelligence called 'AI-Winters'

1995 Support Vector Machines are first introduced

2002 Torch first release (open source library)

2006 Geoffrey Hinton coins 'Deep Learning' [HOT06]

>2006 Companies such as Netflix, Facebook, Microsoft, Google fund projects/prizes in and use machine learning/artificial intelligence

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)
4. Analyze Data (define features, find correlations)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)
4. Analyze Data (define features, find correlations)
5. Prepare Data (transform, normalize, drop)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)
4. Analyze Data (define features, find correlations)
5. Prepare Data (transform, normalize, drop)
6. Evaluate Models (train/test, classify/regress)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)
4. Analyze Data (define features, find correlations)
5. Prepare Data (transform, normalize, drop)
6. Evaluate Models (train/test, classify/regress)
7. Tune Model (cross validation, fine tune parameters)

Machine learning techniques follow a similar workflow.

1. Define Problem (scope, feasability)
2. Gather Data (assumptions, constraints)
3. Pre-process Data (cleanup, drop)
4. Analyze Data (define features, find correlations)
5. Prepare Data (transform, normalize, drop)
6. Evaluate Models (train/test, classify/regress)
7. Tune Model (cross validation, fine tune parameters)
8. Apply model to problems, learn more

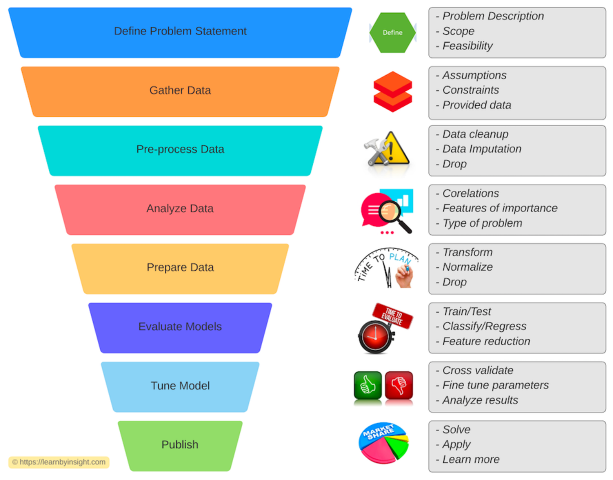

Figure 1: Machine Learning workflow [Mew20]

CONCEPTS

Supervised

Supervised

- Fit model to labelled data (ie. with ’ground truth’)

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans

Supervised

- Fit model to labelled data (ie. with 'ground truth')
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

UNSUPERVISED AND SUPERVISED LEARNING

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)

UNSUPERVISED AND SUPERVISED LEARNING

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)
- Find structure in data (clustering, grouping)

UNSUPERVISED AND SUPERVISED LEARNING

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)
- Find structure in data (clustering, grouping)

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)
- Find structure in data (clustering, grouping)

Semi-supervised

UNSUPERVISED AND SUPERVISED LEARNING

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)
- Find structure in data (clustering, grouping)

Semi-supervised

- Combine partly labeled data with partly unlabeled data

Supervised

- Fit model to labelled data (ie. with ’ground truth’)
- Data is usually obtained experimentally or assigned by humans
- Previously labelled data can serve as testing set

Unsupvervised

- Data does not contain any labels (only inputs)
- Find structure in data (clustering, grouping)

Semi-supervised

- Combine partly labeled data with partly unlabeled data
- Can have huge performance benefits compared to unsupervised learning

TERMS

TERMS

- Classification: Assign datapoints discrete categories (eg. cancerous, non-cancerous). Algorithms are called 'classifiers'.

- Classification: Assign datapoints discrete categories (eg. cancerous, non-cancerous). Algorithms are called ’classifiers’.

  If discrete categories are mutually exclusive, we call them ’classes’, otherwise ’labels’.

- Classification: Assign datapoints discrete categories (eg. cancerous, non-cancerous). Algorithms are called ’classifiers’.

  If discrete categories are mutually exclusive, we call them ’classes’, otherwise ’labels’.
- Regression: Output continuous values (eg. predict free energy of protein system).

- Classification: Assign datapoints discrete categories (eg. cancerous, non-cancerous). Algorithms are called 'classifiers'.

  If discrete categories are mutually exclusive, we call them 'classes', otherwise 'labels'.
- Regression: Output continuous values (eg. predict free energy of protein system).
- Classification problems can also be solved with regression and thresholds/binning.

- Classification: Assign datapoints discrete categories (eg. cancerous, non-cancerous). Algorithms are called 'classifiers'.

  If discrete categories are mutually exclusive, we call them 'classes', otherwise 'labels'.
- Regression: Output continuous values (eg. predict free energy of protein system).
- Classification problems can also be solved with regression and thresholds/binning.
- Clustering: Predict groupings of similar datapoints.

- Loss or Cost function:
  Measure deviation to ground truth in supervised learning.
  Implemented similarly in unsupervised situations.

- Loss or Cost function:
  Measure deviation to ground truth in supervised learning.
  Implemented similarly in unsupervised situations.
- Parameters: Part of the model, will be adjusted by learning process of the model.

- Loss or Cost function:
  Measure deviation to ground truth in supervised learning.
  Implemented similarly in unsupervised situations.
- Parameters: Part of the model, will be adjusted by learning process of the model.
- Hyperparameters: Not part of the model but control learning process (eg. learning rate, number of iterations)

- Loss or Cost function:
  Measure deviation to ground truth in supervised learning.
  Implemented similarly in unsupervised situations.
- Parameters: Part of the model, will be adjusted by learning process of the model.
- Hyperparameters: Not part of the model but control learning process (eg. learning rate, number of iterations)
- Training: describes process of iterative learning and adjusting the parameters of the model to obtain better performance. Minimize the loss/cost-function.

TERMS

- Loss or Cost function:
  Measure deviation to ground truth in supervised learning.
  Implemented similarly in unsupervised situations.
- Parameters: Part of the model, will be adjusted by learning process of the model.
- Hyperparameters: Not part of the model but control learning process (eg. learning rate, number of iterations)
- Training: describes process of iterative learning and adjusting the parameters of the model to obtain better performance. Minimize the loss/cost-function.
- Validation: Use seperate dataset to test model.

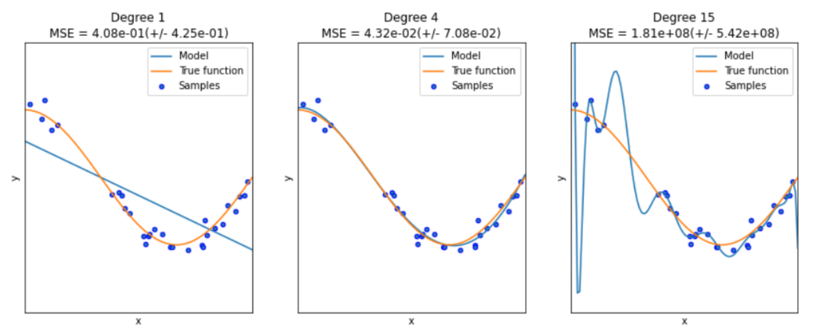

Figure 2: Underfitting, Optimal Fitting and Overfitting [Tri20]

- Inductive Bias: Set of assumptions.

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.

INDUCTIVE BIAS AND VARIANCE

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.
  Lower bias makes fewer assumptions.

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.
  Lower bias makes fewer assumptions.
- Variance: How much does trained model change in response to training on different dataset.

INDUCTIVE BIAS AND VARIANCE

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.
  Lower bias makes fewer assumptions.
- Variance: How much does trained model change in response to training on different dataset.
- We want low bias and low variance.

INDUCTIVE BIAS AND VARIANCE

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.
  Lower bias makes fewer assumptions.
- Variance: How much does trained model change in response to training on different dataset.
- We want low bias and low variance.
- Low bias and low variance often conflict each other.

- Inductive Bias: Set of assumptions.
  Leads it to favour a particular type of solution over others.
  Often programmed in mathematical model.
  Example: Recurrent Neural Networks anticipate sequential dependencies
- Trade-off between bias and variance
  Different inductive biases typically lead to better performance, but higher constraints on the model.
  Lower bias makes fewer assumptions.
- Variance: How much does trained model change in response to training on different dataset.
- We want low bias and low variance.
- Low bias and low variance often conflict each other.
  $\Rightarrow$ Need to balance between them

MACHINE LEARNING TECHNIQUES

BIOLOGY AGAIN

Questions?

[HOT06] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A fast learning algorithm for deep belief nets. Neural Computation, 18(7):1527–1554, July 2006.

[Mew20] Sandeep Mewara. codeproject.com/, 2020.

[Mit97] Tom M. Mitchell. Machine Learning. McGraw-Hill, 1997.

[MP43] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics, 5(4):115–133, December 1943.

[Nil65] Nils J. Nilsson. Learning machines. McGraw-Hill, 1965.

[Tri20] Mayank Tripathi. Underfitting and overfitting in machine learning, 2020.