Professional Certificate in Machine Learning and Artificial Intelligence

Programme Outline

Module 0

Key takeaways: NA

Key activities

- Videos
- Office hours

Module 1: Python Refresher

Key takeaways: NA

Key activities

- Videos
- Jupyter Notebooks for practice
- Office hours

Module 2: Introduction to Machine Learning

Key takeaways

- Identify the fundamental components of and approaches to machine learning problems.
- Differentiate between machine learning and statistics.
- Classify problems along the major dividing lines of the machine learning landscape.
- Apply the ten steps of a typical machine learning project.
- Identify real-world applications of machine learning in a variety of industries.

Key activities

- Videos 2.1–2.9
- Required activity 2.1: Quiz on comparing machine learning and statistics
- Required activity 2.2: Dealing with missing data
- Required activity 2.3: Discussion on identifying a real-world machine learning problem
- Required activity 2.4: Defining key components of a real-world machine learning problem
- Required activity 2.5: Quiz

Module 3: Introduction to Machine Learning

Key takeaways

- Categorise examples of probability versus statistics.
- Calculate absolute, conditional and total probabilities.
- Classify independent versus dependent events.
- Recognise equations that use Bayes' rule correctly.
- Run simulations using random number generating libraries in Python and NumPy.
- Distinguish between discrete and continuous random variables.
- Identify and compute probabilities and values related to binomial distribution.
- Define central limit theorem function values.
- Predict the impact of changes in variables on changes in distributions..

Key activities

- Videos 3.1–3.22
- Required activity 3.1: Quiz on understanding probability
- Required activity 3.2: Jupyter notebook introducing NumPy
- Required activity 3.3: Jupyter notebook on Monte Carlo simulations
- Required activity 3.4: Discussion on real-world applications of discrete and continuous variables
- Required activity 3.5: Quiz on binomial distribution
- Required activity 3.6: Quiz on manipulating normal variables

---

Module 4: Statistics for Machine Learning

Key takeaways

- Conduct a maximum likelihood estimation using existing data.
- Detect outliers in a data set.
- Calculate the regression or correlation coefficients for a data set.
- Compute confidence intervals.
- Analyse the potential consequences of eliminating outliers from a data set.
- Assess the validity of statements about data sets and relations among data.
- Predict outcomes using regression calculations.
- Apply outlier removal techniques and correlations to a business case.

Key activities

- Videos 4.1–4.7
- Required activity 4.1: Jupyter notebook on maximum likelihood estimation
- Required activity 4.2: Jupyter notebook on bootstrapping
- Required activity 4.3: Discussion on the impact of removing outliers

- Required activity 4.4: Analysing the DHL data set
- Required activity 4.5: Analysis of the DHL case study
- Required activity 4.6: Summary quiz

Module 5: Generalisation Theory and the Bias–Variance Tradeoff

Key takeaways

- Identify when it is feasible to draw meaningful conclusions from data.
- Calculate the probability of selecting the correct model from a given number of samples.
- Describe important components of generalisation bounds.
- Select a model that best fits a set of data using a training set and validation set.
- Estimate the fit of a selected model on new, unseen data using a test set.

Key activities

- Videos 5.1–5.9
- Required activity 5.1: Quiz on evaluating assumptions about data
- Required activity 5.2: Quiz on the feasibility of machine learning
- Required activity 5.3: Applying the 'training set–validation set–test set' approach
- Required activity 5.4: Discussion on the feasibility of your machine-learning problem

---

Module 6: Evaluating Predictive Performance

Key takeaways

- Evaluate regression problems using performance measures where the output variable is numerical.
- Evaluate performance in classification problems where the output variable is categorical using a confusion matrix.
- Evaluate the performance of ranking problems using lift charts.
- Identify challenges in different industries that can be addressed by machine learning competitions.

Key activities

- Videos 6.1–6.9
- Required activity 6.1: Training and validation sets
- Required activity 6.2: Building a confusion matrix
- Required activity 6.3: Graphing and interpreting lift charts in Python
- Required activity 6.4: Discussion on machine learning competitions

- Required activity 6.5: Quiz on evaluating predictive performance

---

Module 7: Advanced Topics in Performance Evaluation

Key takeaways

- Identify examples where oversampling might be useful.
- Apply oversampling to classification problems where the binary data of interest is rare.
- Estimate the performance of a given predictor using the k-fold cross-validation algorithm.
- Apply predictive techniques to a multi-variable, real-world data set.

Key activities

- Videos 7.1–7.6
- Required activity 7.1: Discussion on applications of predictive performance
- Required activity 7.2: Assignment on applying oversampling to a classification problem
- Required activity 7.3: Assignment on estimating performance with k-fold cross-validation
- Required activity 7.4: Assignment on applying predictive techniques to real-world data

---

Module 8: Nearest Neighbour Methods

Key takeaways

- Calculate distance functions for the k-nearest neighbours method.
- Convert binary and categorical predictors into numbers and vice versa, taking into consideration whether data is nominal or ordinal.
- Apply normalisation methods to scale data sets.
- Predict and select the value of k for regression and classification problems, using validation and test sets.
- Recognise how the Curse of dimensionality can impact the validity of distance-based methods that use high-dimensional training data.
- Describe real-life applications for the k-nearest neighbours method that take into context the advantages and shortcomings of such methods.

Key activities

- Videos 8.1–8.11
- Required activity 8.1: Predicting plant types

- Required activity 8.2: Converting binary and categorical predictors using Python
- Required activity 8.3: Quiz on the curse of dimensionality
- Required activity 8.4: Jupyter Notebook on applying k-nearest neighbours to a classification problem
- Required activity 8.5: Discussion on applications of the k-nearest neighbours method
- Feedback survey

Module 9: Decision Trees, Part one

Key takeaways

- Demonstrate how a decision tree makes predictions.
- Discuss the difference between conducting splits on categorical and numerical input variables.
- Measure purity in categorical models using entropy and the Gini index.
- Demonstrate how a computer constructs a decision tree.
- Compare strategies for pruning a classification tree.
- Recognise the differences in constructing regression rather than classification trees.
- Apply decision trees within the context of solving a business challenge.

Key activities

- Videos 9.1–9.12
- Required activity 9.1: Calculating the misclassification rate and confusion matrix
- Required activity 9.2: Using entropy and the Gini index to grow a decision tree
- Required activity 9.3: Implementing the computer algorithm by hand
- Required activity 9.4: Case study: Credit Saison
- Required activity 9.5: Discussion on preventing churn in your organisation

Module 10: Decision Trees, Part two

Key takeaways

- Select the most appropriate tree depth for making a prediction.
- Identify important elements of tree ensembles.
- Implement important elements of tree ensembles.
- Recognise the importance of features in tree ensembles.
- Identify real-life applications of decision trees.
- Discuss the concepts of interpretability and fairness
- Compare the functionality of k-nearest neighbours and decision trees.

Key activities
- Videos 10.1–10.6
- Required activity 10.1: Selecting tree depth
- Required activity 10.2: Jupyter Notebook on features in tree ensembles
- Required activity 10.3: Discussion on real-life applications of tree ensembles
- Required activity 10.4: How might data sets contain biases?
- Required activity 10.5: Comparing k-nearest neighbours and decision trees

---

Module 11: Naïve Bayes

Key takeaways
- Use Bayes' theorem to calculate conditional probabilities.
- Discuss examples of situations where you might want to predict a probability rather than an actual value.
- Describe important components of Bayes' theorem.
- Build a simple naive Bayes classifier.
- Convert numbers into categorical predictors.
- Discuss real-life applications of naive Bayes.

Key activities
- Videos 11.1–11.11
- Required activity 11.1: Use Bayes' theorem to calculate the probability of falling sick
- Required activity 11.2: Classifying spam vs ham emails
- Required activity 11.3: Discussion on real-life applications
- Required activity 11.4: Use a text message data set to calculate 'spam' vs. 'not spam'

Module 12: Bayesian Optimisation

Key takeaways
- Identify the parameters in machine learning algorithms and the most common surrogate methods used for tuning.
- Represent exploration and exploitation in Bayesian optimisation.
- Analyse the trade-offs between exploration and exploitation for specific applications.
- Determine when continued parameter tuning is no longer worthwhile.
- Describe the purpose of using surrogate models, their applications and common pitfalls.

- Identify a codebase for machine learning competitions.

Key activities
- Videos 12.1–12.7
- Required activity 12.1: Identifying parameters in regression trees
- Required activity 12.2: Jupyter Notebook on Bayesian optimisation
- Required activity 12.3: Discussion on real-world applications for your industry
- Required activity 12.4: Reflection on strategy for capstone competition

---

Module 13: Logistic Regression

Key takeaways
- Distinguish logistic regression from linear regression in terms of its categorical output and its best use cases for binary classification.
- Describe how adjustments to parameters impact the shape of logistic functions.
- Determine the optimal number of statistically relevant predictors to add to a logistic regression function.
- Apply a maximum likelihood method to fit a logistic regression to a real-life data set.
- Compare the advantages and disadvantages of applying logistic regression relative to other classification methods.
- Identify a codebase for machine learning competitions.

Key activities
- Videos 13.1–13.7
- Required activity 13.1: Quiz on linear vs logistic regression
- Required activity 13.2: Jupyter Notebook on setting up and optimising a logistic function
- Required activity 13.3: Discussion on logistic regression and other classification methods
- Required activity 13.4: Submit query to capstone competition

Module 14: Support Vector Machines

Key takeaways
- Describe how support vector machines separate data using maximum margin hyperplanes.
- Identify popular kernel lifting classes, explain the implications of different error rates between them and select a kernel type appropriate to a given project.
- Explain why soft-margin classifiers are better-suited for handling data with outliers.
- Describe how support vector machines can be applied to classification problems

with more than two outcomes.
- Apply SVMs to a high-dimensional machine learning problem.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 14.1–14.7
- Required activity 14.1: Quiz on case study
- Required activity 14.2: Jupyter Notebook on applying support vector machines
- Required activity 14.3: Discussion on real-world applications of SVMs
- Required activity 14.4: Refine submission for the capstone competition

---

Module 15: Unsupervised Learning

Key takeaways
- Extend the concept of distance to clusters.
- Conduct a hierarchical clustering and draw a dendrogram.
- Provide examples that prevent the best possible clustering.
- Conduct a k-means clustering.
- Identify practical issues that arise with clustering and how to manage them.
- Discuss the real-life applications of clustering.
- Apply k-means and hierarchical clustering in Python to real-life data.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 15.1–15.8
- Required activity 15.1: Jupyter Notebook on conducting a hierarchical clustering
- Required activity 15.2: Jupyter Notebook on conducting a k-means clustering
- Required activity 15.3: Jupyter Notebook on applying clustering to a real-life data set
- Required activity 15.4: Discussion on real-world applications of clustering analysis
- Required activity 15.5: Refine submission for the capstone competition

---

Module 16: Principal Component Analysis

Key takeaways
- Describe the value of PCA for exploratory data analysis and dimensionality reduction.
- Interpret the variance of data points when plotting components in two dimensions.

- Determine the optimal number of components to use for analysis.
- Identify how the direction and location of principal components are calculated.
- Prepare data for PCA by mean centring and variable scaling.
- Apply the kernel trick in PCA of non-linear data.
- Recognise advantages of applying PCA for problems where interpretability is a priority.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 16.1–16.9
- Required activity 16.1:Jupyter Notebook on applying PCA
- Required activity 16.2: Quiz on PCA
- Required activity 16.3: Discussion on real-world applications of PCA
- Required activity 16.4: Reflection on machine learning methods that impact your field
- Required activity 16.5: Refine submission for capstone competition
- Required activity 16.6: Quiz on manipulating normal variables

---

Module 17: Introduction to Deep Learning

Key takeaways
- Evaluate literature on deep learning in order to apply it to a particular industry or use case.
- Identify the building blocks used for developing real-life deep learning frameworks.
- Describe the key terms and concepts used in deep learning and artificial intelligence.
- Recognise business applications for real-life neural networks operating at a large scale of data.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 17.1–17.7
- Required activity 17.1: Identifying a human benchmark
- Required activity 17.2: Quiz on the building blocks of deep learning
- Required activity 17.3: Constructing a boundary
- Required activity 17.4: Discussion on potential applications in your industry
- Required activity 17.5: Refine submission for the capstone competition

---

Module 18: Neural Networks

Key takeaways
- Explain the architecture of a neural network.
- Use the gradient descent algorithm and the backpropagation technique.
- Determine when deep learning is a feasible/realistic solution for a data-based problem.
- Develop a neural network using the basic building blocks.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 18.1–18.9
- Required activity 18.1: Jupyter Notebook on developing neural networks
- Required activity 18.2: Jupyter Notebook on gradient descent and backpropagation
- Required activity 18.3: Reflection on when deep learning is a feasible solution
- Required activity 18.4: Refine submission for the capstone competition

---

Module 19: Hyperparameters

Key takeaways
- Dissect a simple neural network using the PyTorch optimisation framework.
- Evaluate the trade-offs of employing deep neural networks in your machine learning projects.
- Refine an existing neural network for a specific application.
- Develop intuition on the structure and components of a neural network.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 19.1–19.7
- Required activity 19.1: Quiz on hyperparameters
- Required activity 19.2: Jupyter Notebook on using PyTorch
- Required activity 19.3: Discussion on the trade-offs of utilising deep learning in your industry
- Required activity 19.4: Refine submission for capstone competition

---

Module 20: Transparency and Interpretability

Key takeaways
- Define transparency and interpretability.
- Recognise bias inherent in data.
- Develop a datasheet for a dataset.
- Recognise bias in the creation of machine learning models.
- Identify ways to benchmark models against reporting standards.
- Identify trade-offs between explainability or interpretability.
- Build an interpretable decision tree.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 20.1–20.7
- Required activity 20.1: Developing a datasheet for a dataset
- Required activity 20.2: Analysing a model card
- Required activity 20.3: Discussion on evaluating real-world trade-offs
- Required activity 20.4: Building an interpretable decision tree
- Required activity 20.5: Refine submission for the capstone competition

---

Module 21: Convolutional Neural Networks

Key takeaways
- Describe the anatomy of convolutions and how they work.
- Describe how convolutions are used for computer vision applications.
- Analyse the trade-off between different network architectures as they relate to domain applications.
- Describe how a network architecture would be applied to your industry use case.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 21.1–21.5
- Required activity 21.1: Quiz on structure of CNNs
- Required activity 21.2: Analysing a CNN example
- Required activity 21.3: Discussion on CNNs in your industry
- Required activity 21.4: Refine submission for capstone competition

---

Module 22: Biological Basis for Convolutional Neural Networks

Key takeaways

- Dissect the LeNet-5 convolutional network architecture.
- Customise convolutional neural networks in PyTorch.
- Build a computer vision system that allows you to classify images in different categories.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 22.1–22.7
- Required activity 22.1: Discussion on the biological basis of CNNs
- Required activity 22.2: Customising a CNN in PyTorch
- Required activity 22.3: Classifying faces
- Required activity 22.4: Refine submission for the capstone competition

---

Module 23: Reinforcement Learning

Key takeaways
- Identify the main components of a reinforcement learning problem.
- Describe the trade-off between exploration and exploitation when trying to maximise for long-term rewards.
- Describe relative advantages and potential applications for model-based reinforcement learning frameworks.
- Apply general reinforcement learning concepts to the multi-armed bandits model.
- Differentiate between model-based and model-free approaches to reinforcement learning.
- Refine a codebase for machine learning competitions.

Key activities
- Videos 23.1–23.16
- Required activity 23.1: Quiz on reinforcement learning concepts
- Required activity 23.2: Applying reinforcement learning to multi-armed bandits
- Required activity 23.3: Discussion on reinforcement learning within your industry
- Required activity 23.4: Refine your capstone submission

Module 24: Hyperparameter Tuning

Key takeaways
- Define the challenges in hyperparameter tuning
- Identify the consequences of having bad hyperparameters

- Describe the methods for hyperparameter tuning and appropriate use cases
- Synthesise key contributions from current literature on hyperparameter tuning
- Brainstorm new ideas for hyperparameter tuning and justify those ideas
- Refine a codebase for machine learning competitions.

Key activities
- Videos 24.1–24.4
- Required activity 24.1: Discussion on the synthesis of current literature
- Required activity 24.2: Reflection on the future of hyperparameter tuning
- Required activity 24.3: Final capstone submission
- Required activity 24.4: Preparing for the final project

Capstone

Key takeaways
- Optimise an existing or original code base to run against a test function.
- Discuss strategies for succeeding in a Bayesian optimisation competition.
- Reflect on the application of machine learning and artificial intelligence to career goals.
- Apply Bayesian optimisation to a real-life data challenge in a specific industry.

Key activities
- Videos 25.1 and 25.2
- Required activity 25.1: Retrospective on the capstone competition
- Required activity 25.2: Discussion on successful optimisation strategies
- Required activity 25.3: Portfolio project on optimising a model for real-life data