



# ADACS

ASTRONOMY DATA AND COMPUTING SERVICES

## Astroinformatics school - "Rise of the machines"



4 to 6 February 2019

Presented by Rebecca Lange and Dan Marrable

# Reference sheets



Home Installation Documentation Examples

Google Custom Search Search x



## scikit-learn

Machine Learning in Python

- Simple and efficient tools for data mining and data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

## Classification

Identifying to which category an object belongs to.

**Applications:** Spam detection, Image recognition.

**Algorithms:** SVM, nearest neighbors, random forest, ... — Examples

## Regression

Predicting a continuous-valued attribute associated with an object.

**Applications:** Drug response, Stock prices.

**Algorithms:** SVR, ridge regression, Lasso, ... — Examples

## Clustering

Automatic grouping of similar objects into sets.

**Applications:** Customer segmentation, Grouping experiment outcomes

**Algorithms:** k-Means, spectral clustering, mean-shift, ... — Examples

## Dimensionality reduction

Reducing the number of random variables to consider.

**Applications:** Visualization, Increased efficiency

**Algorithms:** PCA, feature selection, non-negative matrix factorization. — Examples

## Model selection

Comparing, validating and choosing parameters and models.

**Goal:** Improved accuracy via parameter tuning


**Modules:** grid search, cross validation, metrics. — Examples

## Preprocessing

Feature extraction and normalization.

**Application:** Transforming input data such as text for use with machine learning algorithms.

**Modules:** preprocessing, feature extraction. — Examples



## Keras Documentation

Search docs

Home

You have just found Keras.

Guiding principles

Getting started: 30 seconds to Keras

Installation

Configuring your Keras backend

Support

Why this name, Keras?

Why use Keras

GETTING STARTED

Guide to the Sequential model

Guide to the Functional API

FAQ

MODELS

About Keras models

Sequential

Model (functional API)

LAYERS

About Keras layers

Core Layers

GitHub

Next »

Docs » Home

[Edit on GitHub](#)

## Keras: The Python Deep Learning library



### You have just found Keras.

Keras is a high-level neural networks API, written in Python and capable of running on top of [TensorFlow](#), [CNTK](#), or [Theano](#). It was developed with a focus on enabling fast experimentation. *Being able to go from idea to result with the least possible delay is key to doing good research.*

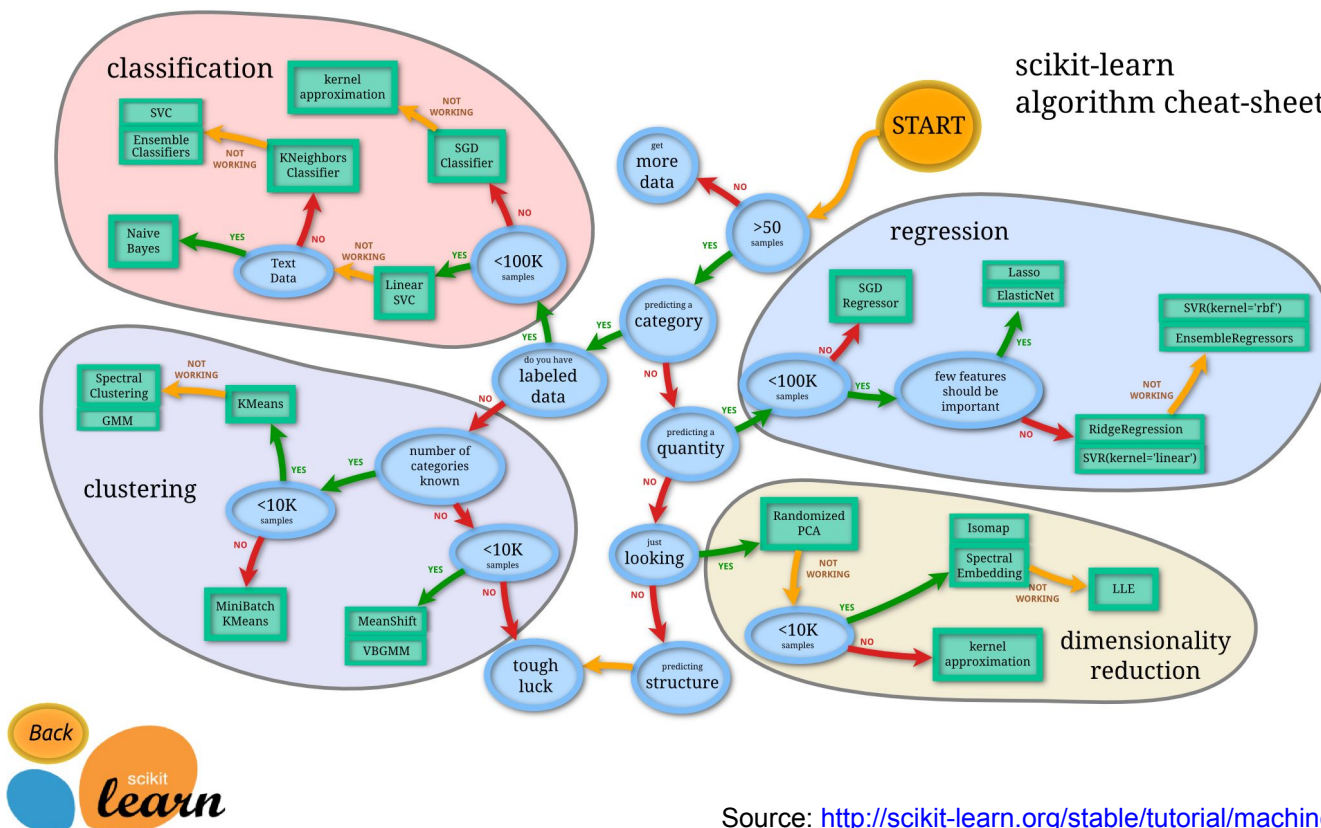
Use Keras if you need a deep learning library that:

- Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).
- Supports both convolutional networks and recurrent networks, as well as combinations of the two.
- Runs seamlessly on CPU and GPU.

Read the documentation at [Keras.io](#).

Keras is compatible with: **Python 2.7-3.6**.

# Reference sheets



# Reference sheets

## Comparison of supervised machine learning techniques

Factors	Decision Trees	Neural Networks	Naïve Bayes	kNN	SVM	Rule Learners
General accuracy	**	***	*	**	****	**
Learning speed	***	*	****	****	*	**
Classification speed	****	****	****	*	****	****
Tolerance to missing values	***	*	****	*	**	**
Tolerance to irrelevant features	***	*	**	**	****	**
Tolerance to redundant features	**	**	*	**	***	**
Tolerance to highly related features	**	***	*	*	***	**
Dealing with discrete, binary and continuous features	****	***	***	***	**	***
Tolerance to noise	**	**	***	*	**	*
Dealing with model overfitting	**	*	***	***	**	**
Attempts for incremental learning	**	***	****	****	**	*
Explanation of classification	****	*	****	**	*	****
Model parameter handling	***	*	****	***	*	***

\* kNN = k-Nearest Neighbours

\* SVM = Support Vector Machines

# Reference sheets

- Supervised learning
  - [http://scikit-learn.org/stable/supervised\\_learning.html#supervised-learning](http://scikit-learn.org/stable/supervised_learning.html#supervised-learning)
- Performance metrics
  - <http://scikit-learn.org/stable/modules/classes.html#module-sklearn.metrics>
- Feature selection
  - [http://scikit-learn.org/stable/modules/feature\\_selection.html](http://scikit-learn.org/stable/modules/feature_selection.html)
- Model selection (cross validation, learning curves, hyperparameter tuning)
  - [http://scikit-learn.org/stable/modules/classes.html#module-sklearn.model\\_selection](http://scikit-learn.org/stable/modules/classes.html#module-sklearn.model_selection)
- Clustering
  - <http://scikit-learn.org/stable/modules/clustering.html#clustering>
- Dimensionality reduction
  - <http://scikit-learn.org/stable/modules/classes.html#module-sklearn.decomposition>
- Deep Learning with Python
  - <https://github.com/fchollet/deep-learning-with-python-notebooks>

# Reference sheets

- Anomaly detection
  - [http://scikit-learn.org/stable/modules/outlier\\_detection.html](http://scikit-learn.org/stable/modules/outlier_detection.html)
- Time series
  - <http://machinelearningmastery.com/arma-for-time-series-forecasting-with-python/>
  - <https://machinelearningmastery.com/time-series-prediction-lstm-recurrent-neural-networks-python-keras/>
  - [http://www.statsmodels.org/dev/generated/statsmodels.tsa.arima\\_model.ARIMA.html](http://www.statsmodels.org/dev/generated/statsmodels.tsa.arima_model.ARIMA.html)
- ML blogs:
  - <https://www.analyticsvidhya.com/blog/2015/08/common-machine-learning-algorithms/>
  - <http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/>

# Reference sheets

- Online courses
  - Machine learning
    - <https://www.coursera.org/learn/machine-learning>
    - <https://www.udacity.com/course/intro-to-machine-learning--ud120>
  - Deep learning
    - <https://www.coursera.org/specializations/deep-learning>
    - <https://www.udacity.com/course/deep-learning--ud730>
    - <http://course.fast.ai/>
- “Playgrounds”
  - Tensorflow playground <https://playground.tensorflow.org/>
  - Visualising the mnist data going through a network <http://scs.ryerson.ca/~aharley/vis/>