Read data from xlrd

Ethans
Learn from experts

- **myWorkBook.sheets()** – Returns the object of individual sheet to work on.
- **myWorkBook.sheet_names()** – Returns the list of all sheets.
- **myWorkBook.sheet_by_index()** – Returns the object of individual sheet based on index value.
- **myWorkBook.nsheets** – An integer containing the number of sheets in a workbook.
- **Sheet.name** – Returns the sheet name.
- **sheet.nrows** and **sheet.ncols** – Contain the number of rows and columns in a worksheet.
- **cellname** and **colname** – Defined functions in xlrd that provide the cell name and colname.

Xlwt – write XL files

Ethans
Learn from experts

```
import xlwt

book = xlwt.Workbook()
sheet1 = book.add_sheet('names')

# Add data using row and col
sheet1.write(0,0,'A1')
sheet1.write(0,1,'B1')

# Add data using row object
row1 = sheet1.row(1)
row1.write(0,'A2')
row1.write(1,'B2')

# Add data using row ID
sheet1.row(2).write(0,'A3')
sheet1.row(2).write(1,'B3')
sheet1.flush_row_data()

book.save('writeToXL.xls')
```

Xlutils – Append Data to XL file

Ethan's Learn From experts

```
from xlutils.copy import copy
import xlrd

book = xlrd.open_workbook("Sample.xls")
copy_book = copy(book)              #creates a writeable copy
sheet1 = copy_book.get_sheet(0)  # get a first sheet

col = 3
for row in range(1):
    # Write the data to row, column
    sheet1.write(row,col, 10)
    sheet1.write(row,col + 1, 10)

copy_book.save("Sample.xls")
```


Web scraping Example using urllib

Ethan's Learn From experts

```
import re, urllib

def get_quote(symbol):
  base_url = 'http://finance.google.com/finance?q='
  content = urllib.urlopen(base_url + symbol).read()
  find_q = re.search(r'\<span\sid="ref_\d+.*">(.+)<', content)
  if find_q:
    quote = find_q.group(1)
  else:
    quote = 'no quote available for: %s' % symbol
  return quote

Companys = ['google', 'ibm', 'microsoft', 'apple', 'nvidia']
for i,value in enumerate(Companys):
    print '%s --> %s' %  (Companys[i],get_quote(value))
```

Web scraping Example using bs4

Ethans
Learn from experts

```
from bs4 import BeautifulSoup
import requests, re

myurl = "http://finance.yahoo.com/q/op?s=" + symbol + "+Options"
html = requests.get(myurl).content
soup = BeautifulSoup(html, "html.parser")
print soup.find('span', attrs={'class':'time_rtq_ticker'}).text
```


Some Common important modules

Ethans
Learn from experts

1. Requests. The Requests module is a an elegant and simple HTTP library for Python It's a must have for every python developer.
2. Scrapy. If you are involved in webscraping then this is a must have library for you.
3. wxPython is a wrapper for the cross-platform GUI API (often referred to as a 'toolkit) for the Python programming language
4. Pillow. A friendly fork of PIL (Python Imaging Library). It is more user friendly than PIL and is a must have for anyone who works with images.
5. SQLAlchemy. A database library. Many love it and many hate it. It includes dialects for SQLite, Postgresql, MySQL, Oracle, MS-SQL, Firebird, Sybase and others,
6. BeautifulSoup. This parse xml and html library very useful for beginners.
7. Twisted. The most important tool for any network application developer. It has a very beautiful api and is used by a lot of famous python developers.
8. NumPy. How can we leave this very important library ? It provides some advance math functionalities to python.
9. SciPy. When we talk about NumPy then we have to talk about scipy. It is a library of algorithms and mathematical tools for python and has caused many scientists to switch from ruby to python.

Some Common important modules

Ethans
Learn from experts

1. matplotlib. A numerical plotting library. It is very useful for any data scientist or any data analyzer.
2. Pygame. Which developer does not like to play games and develop them ? This library will help you achieve your goal of 2d game development.
3. Pyglet. A 3d animation and game creation engine. This is the engine in which the famous python port of minecraft was made
4. pyQT. A GUI toolkit for python. It is my second choice after wxpython for developing GUI's for my python scripts.
5. nltk. Natural Language Toolkit – I realize most people won't be using this one, but it's generic enough. It is a very useful library if you want to manipulate strings. But it's capacity is beyond that. Do check it out.
6. nose. A testing framework for python. It is used by millions of python developers. It is a must have if you do test driven development.

Slide 247

www.ethans.co.in

Objective – Module 11

Ethans
Learn from experts

Introduction to Machine Learning

- What is Machine Leaning?
- Areas of Implementation of Machine Learning,
- Why companies prefer ML
- Major Classes of Learning Algorithms
- Supervised vs Unsupervised Learning, Learning
- Why Numpy?
- Learning Numpy and Scipy,
- Basic plotting using Matplotlib
- Algorithms using Skikit learn

Slide 248

What is Machine Learning?

Ethans
Learn From experts

Machine learning was defined in 1959 by Arthur Samuel as the "field of study that gives computers the ability to learn without being explicitly programmed." This means imbuing knowledge to machines without hard-coding it. – via Wikipedia

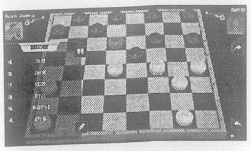
Machine learning is a method of data analysis that automates analytical model building. Using algorithms that iteratively learn from data, machine learning allows computers to find hidden insights without being explicitly programmed where to look. – via SAS

Method of teaching computers or programs/scripts to make and improve predictions or behaviours based on some training data. Machine learning is a *huge* field, with hundreds of different algorithms for solving different stats problems – via Ethans

Slide 249

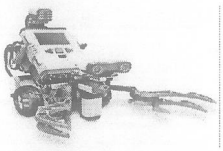
Fields of study

Ethans
Learn From experts

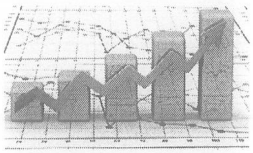


Samuel created a computer checkers in 1959, and seminal research on machine learning beginning in 1949. He thought that teaching computers to play games was very fruitful for developing tactics appropriate to general problems.

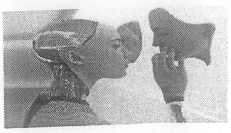
Gaming and AI



Robotics and self driven cars



Forecasting & Bioinformatics



Slide 250

www.ethans.co.in

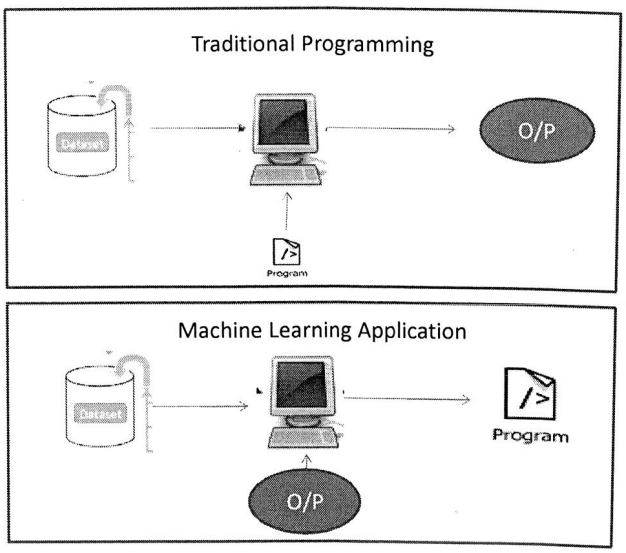
Why companies using Machine learning?

Ethans
Learn From experts

- Online recommendation engines like those from Amazon, snap deal and flip kart. Machine learning applications help them to sell products.
- Got to know the product sentiments from the market from Twitter and Facebook, Machine learning applications help them to analyse the market response.
- Remember face recognition and thumb impression software's for laptop and mobile, Machine learning applications help them to build cool products.
- Fraud detection on time, Machine learning applications help them to build secure products.
- What will be the weather forecast for tomorrow? , Machine learning applications help them to build forecasting products.
- Looking for the automated engine for routing the traffic in the cluster nodes? , Machine learning applications help them to build robotics products.

Comparison of two programming's

Ethans
Learn From experts



How is machine learning used today?

Ethans
Learn from experts

Ever wonder how an online retailer provides nearly instantaneous offers for other products that may interest you? Or how lenders can provide near-real-time answers to your loan requests? Many of our day-to-day activities are powered by machine learning algorithms, including:

- Fraud detection.
- Web search results.
- Real-time ads on web pages and mobile devices.
- Text-based sentiment analysis.
- Credit scoring and next-best offers.
- Prediction of equipment failures.
- New pricing models.
- Network intrusion detection.
- Pattern and image recognition.
- Email spam filtering.

Via: SAS website

Slide 253

www.ethans.co.in

Type of learning in ML

Ethans
Learn from experts

- Supervised learning
- Unsupervised learning
- Semi-supervised
- Reinforcement learning

Two of the most widely adopted machine learning methods are **supervised learning** and **unsupervised learning**. Most machine learning – about 70 percent – is supervised learning. Unsupervised learning accounts for 10 to 20 percent. Semi-supervised and reinforcement learning are two other technologies that are sometimes used. Via SAS

Slide 254

www.ethans.co.in

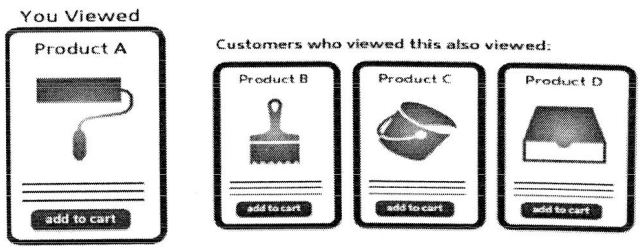
Supervised learning

Ethans
Learn From experts

Supervised learning algorithms are trained using labeled training dataset. The training data consist of a set of training examples.

In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (also called the supervisory signal).

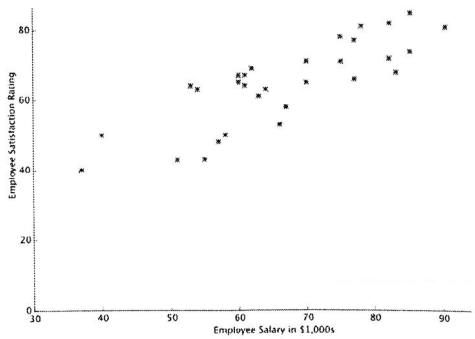
A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples



Recommendation Engine

Examples of Supervised Learning

Ethans
Learn From experts



Linear Regression - Problem

Regression

Ethans
Learn from experts

Regression example

Predict House Pricing:
Based on: House size, floor Size, area, features, Builder etc

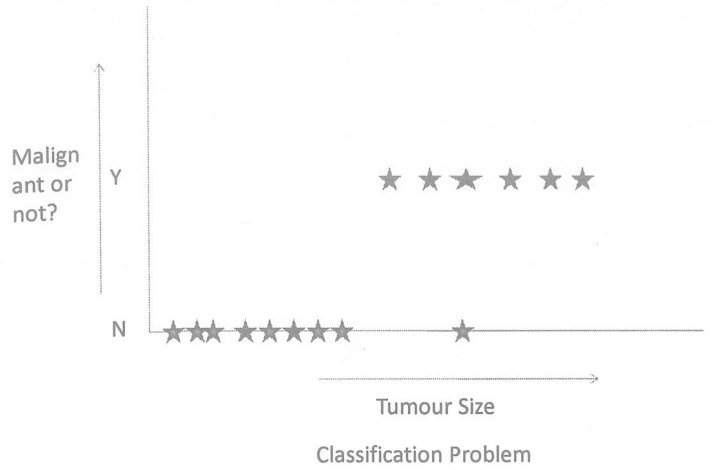
Predict Body BMI:
Based on: Sex, height, weight etc

Predict crop yields:
Based on : Soil conditions, rail fall in an areas, Fertilizers used, water level etc

Predict Life expectancy:
Based on: Eating Pattern, exercises pattern, smoker, drinker , disease status, forefather history etc.

Examples of Supervised Learning

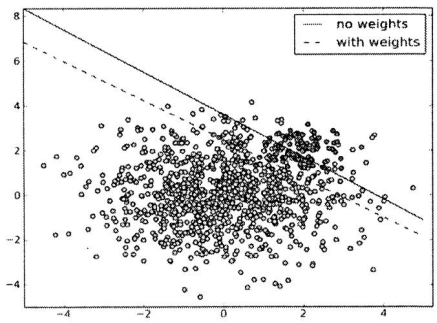
Ethans
Learn from experts



Classification Problem

Examples of Supervised Learning

Ethans
Learn From experts



SVM Algorithm

Slide 259

www.ethans.co.in

Classification

Ethans
Learn From experts

text categorization - (e.g., spam filtering)

fraud detection – Finance, anti virus etc

optical character recognition – Handwriting recognition

machine vision - (e.g., palm detections, thumb impression face detections)

natural-language processing - (e.g., spoken language understanding)

market segmentation - (e.g. predict if customer will respond to promotion)

Bioinformatics - (e.g., classify proteins according to their function)

Slide 260

www.ethans.co.in

Building accurate - Classification

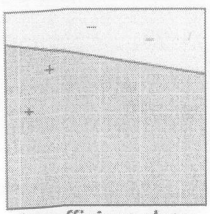
Ethans
Learn From experts

- Get enough training set examples
- It should good in performance on training set
- Classifier should not be not too "complex" to predict
- Classifiers should be "as simple as possible, but no simpler"
- "simplicity" closely related to prior expectations
- In general cases get two label of the classifiers
- Test it many times to get the expected result before finalizing.

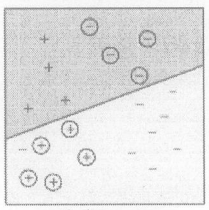
Measure "complexity" by:
- number bits needed to write down
- number of parameters

Bad Classifiers

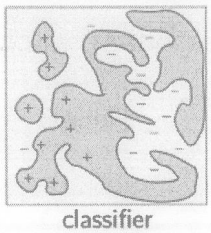
Ethans
Learn From experts



insufficient data



training error too high



classifier too complex

Unsupervised Learning

Ethans
Learn from experts

Input data is not labelled and does not have a known result. Data is organised to form cluster of data.

A model is prepared by creating structures present in the input data.

This may be to extract general rules. It may through a mathematical process to systematically reduce redundancy, or it may be to organize data by similarity.

Example algorithms include: the k-Medians, Expectation Maximisation (EM), Hierarchical Clustering and k-Means.

Unsupervised example:

Ethans
Learn From experts

IPL 9, Qualifier 1: AB de Villiers' 'Superman' knock helps RCB storm ...
Zee News - 8 hours ago
With the spectacular win, RCB are into their third **IPL** final and one step away from their maiden title. Lions on other hand will have another ...

IPL: AB de Villiers pulls off a heist, RCB reach third final
Times of India - 9 hours ago

IPL 9 RCB vs GL: AB De Villiers stands tall amidst ruins, Bangalore ...
Deccan Chronicle - 7 hours ago

IPL 2016, Highlights: RCB vs GL - De Villiers, Abdulla Take Royal ...
NDTVSports.com - 10 hours ago

IPL: De Villiers guides RCB to **IPL** 2016 final, beat Gujarat by 4 wickets
Hindustan Times - 13 hours ago

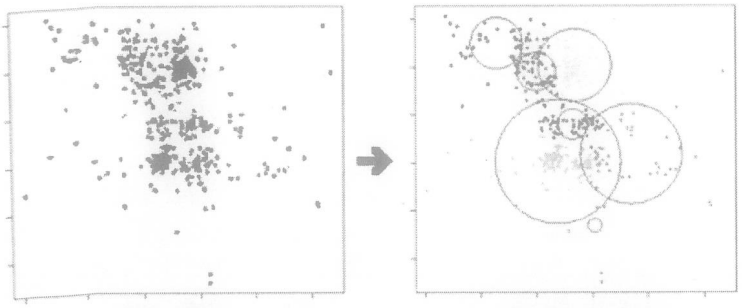
Daniel Vettori and Bangalore into **IPL** final thanks to stars de Villiers ...
Opinion - Stuff.co.nz - 3 hours ago

Times of India | Deccan Chro... | NDTVSports... | Hindustan Tim... | News18 | ABP Live

Explore in depth (675 more articles)

Unsupervised example:

Ethans
Learn From experts



NumPy

Ethans
Learn From experts

NumPy is an extension to the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays

To install NumPy run:
python setup.py install

To perform an in-place build that can be run from the source folder run:
python setup.py build_ext --inplace

The NumPy build system uses distutils and numpy.distutils. setuptools is only used when building via pip or with python setupegg.py.

NumPy performance Test

Ethans
Learn from experts

```
__author__ = 'Ethans'

import timeit

# Sum of 10000 number from Python
py_sec =timeit.timeit
('sum(x*x for x in xrange(10000))',number=10000)

# Sum of 10000 number using numpy
np_sec = timeit.timeit
('na.dot(na)',setup="import numpy as np; na=np.arange(10000)", number=10000)

print("Normal Python: %f sec" %py_sec)
print("Through NumPy: %f sec" %np_sec)
```

```
>>>
Normal Python: 22.872751 sec
Through NumPy: 0.176976 sec
```

Slide 267

www.ethans.co.in

Getting Started with Numpy

Ethans
Learn From experts

```
>>> # Importing Numpy module
>>> import numpy
>>> import numpy as np
```

- Arrays are the central feature of NumPy.
- Arrays in Python are similar to lists in Python, the only difference being the array elements should be of the same type

```
import numpy as np
a = np.array([1,2,3], float)    # Accept two arguments list and type
print a                         # Return array([ 1.,  2.,  3.])
print type(a)                   # Return <type 'numpy.ndarray'>
print a[2]                      # 3.0
print a[1:]                     # array([ 2.,  3.])
print a[:2]                     # array([ 1.,  2.])
```

Slide 268

www.ethans.co.in

Two Dimensional Array

Ethans
Learn from experts

```
import numpy as np
a = np.array([[1,2,3], [4,5,6]], int) # Accept two arguments list and type
>>> a[0][0]   # Return 1
>>> a[0,0]    # Return 1
>>> a[1,2]    # Return 6
>>> a[1:]     # Return array([[4, 5, 6]])
>>> a[1,:]    # Return array([4, 5, 6])
>>> a[:2]     # Return array([[1, 2, 3],[4, 5, 6]])
>>> a[:,2]    # Return array([3, 6])
>>> a[:,1]    # Return array([2, 5])
>>> a[:,0]    # Return array([1, 4])
```

www.ethans.co.in

Slide 269

Two Dimensional Array

Ethans
Learn from experts

```
import numpy as np
a = np.array([[1,2,3], [4,5,6]], int)   # Accept two arguments list and type
>>> a.shape                             # Return the dimension of a i.e. (2,3)
>>> a.dtype                             # Return the type of array i.e. dtype('int32')
>>> b = a.reshape(3,2)                  # Return the new shape of array
>>> b.shape                             # Return (3,2)
>>> len(a)                              # Return length of a
>>> 2 in a                              # Check if 2 is available in a
>>> b.tolist()                          # Return [[1, 2], [3, 4], [5, 6]]
>>> list(b)                             # Return [array([1, 2]), array([3, 4]), array([5, 6])]
>>> c = b
>>> b[0][0] = 10
>>> c                                   # Return array([[10,  2], [ 3,  4], [ 5,  6]])
```

www.ethans.co.in

Slide 270

Two Dimensional Array

Ethans
Learn from experts

```
>>> b                                  # Return array([[10, 2],[ 3, 4],[ 5, 6]])
# Convert the multidimensional array to single dimension
>>> b.flatten()                        # Return array([10, 2, 3, 4, 5, 6])


# Concatenating two arrays
>>> a = np.array([1,2,3])
>>> b = np.array([4,5,6])
>>> np.concatenate((a,b))              # Returns array([1, 2, 3, 4, 5, 6])


# arange function and ones function
>>> a = np.arange(5, dtype=float)      # Return array([ 0., 1., 2., 3., 4.])
>>> b = np.ones((3,2), dtype=float)    # Return array([[ 1., 1.], [ 1., 1.], [ 1., 1.]])
```


Update Array

Ethans
Learn from experts

```
# Assign the new values
------------------------
>>> c = np.array([[1, 2, 3],[4, 5, 6],[7, 8, 9]])
>>> c[0,0] = 10
>>> c
array([[10,  2,  3],
       [ 4,  5,  6],
       [ 7,  8,  9]])
>>> c[:,2] = [10, 10, 10]
>>> c
array([[10,  2, 10],
       [ 4,  5, 10],
       [ 7,  8, 10]])
>>> c[0, :] = [10, 10, 10]
>>> c
array([[10, 10, 10],
       [ 4,  5, 10],
       [ 7,  8, 10]])
>>> c.shape
```

Standard Maths functions

Ethans Learn from experts

```
# 'max', 'mean', 'min', sum, std functions
>>> np.arange(100).max()
99
>>> np.arange(100).mean()
49.5
>>> np.arange(100).min()
0
>>> np.arange(100).sum()
4950
>>> a.std()
2.8722813232690143
>>> a.all()
False
>>> a.any()
True
>>> a.nonzero()
(array([1, 2, 3, 4, 5, 6, 7, 8, 9]),)
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
```

www.ethans.co.in

Slide 273

Copy of arrays

Ethans Learn from experts

```
# Refrential operation also valid here
>>> a = np.array([1,2,3])
>>> a
array([1, 2, 3])
>>> b = a
>>> b[0] = 10
>>> a[2] = 20
>>> a
array([10,  2, 20])
>>> b
array([10,  2, 20])
```

```
# Base copy
>>> a
array([10,  2, 20])
>>> d = a.view()
>>> d
array([10,  2, 20])
>>> a[0] = 20
>>> d[1] = 30
>>> a
array([20, 30, 20])
>>> d
array([20, 30, 40])
>>> a is d
False
>>> d.base is a
True
```

```
# Shallow copy
>>> a
array([10,  2, 20])
>>> c = a.copy()
>>> c
array([10,  2, 20])
>>> a
array([10,  2, 20])
>>> c[0] = 20
>>> c
array([20,  2, 20])
>>> a
array([10,  2, 20])
```

www.ethans.co.in

Slide 274

List vs Array

Ethans
Learn From experts

```
# List operations
>>> list1 = [1,2,3]
>>> list2 = [4,5,6]

>>> list1 + list2
[1, 2, 3, 4, 5, 6]

>>> list2 * 2
[4, 5, 6, 4, 5, 6]
```

```
>>> import numpy as np
>>> a1 = np.array([1,2,3])
>>> a2 = np.array([4,5,6])
>>> a1 + a2
array([5, 7, 9])
>>> a1 * a2
array([ 4, 10, 18])
>>> a2 / a1
array([4, 2, 2])
>>> a1 ** a2
array([  1,  32, 729])
```


Boolean Opr

Ethans
Learn From experts

```
>>> a = np.arange(10)
>>> a
array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
>>>
>>> div = a % 2 == 0
>>> div
array([ True, False,  True, False,  True, False,  True, False,  True, False],
>>> a[div]
array([0, 2, 4, 6, 8])
```

Maths Functions

Ethans
Learn from experts

```
# Mathematical functions
>>> c
array([[1,  2, 4],
       [5,  7, 8]])
>>> c.transpose()
array([[1,  5],
       [2,  7],
       [4,  8]])
>>> np.invert(c)
array([[-2, -3, -5],
       [-6, -8, -9]])
>>> np.eye(3)
array([[ 1.,  0.,  0.],
       [ 0.,  1.,  0.],
       [ 0.,  0.,  1.]])
>>> c = np.array([[1,2], [2,3]])
>>> np.dot(c, c) # Returns dot product array([[ 5,  8],[
>>> a = np.array([1,2,3,4,5,6,7,8,9])
>>> np.median(a) # Return 5.0
>>> np.corrcoef(a) # Return 1.0
```

www.ethans.co.in

Slide 277

Help Page

Ethans
Learn from experts

```
# Documentaion of any fucntion
>>> np.info(np.sum)
 sum(a, axis=None, dtype=None, out=None, keepdims=False)

Sum of array elements over a given axis.

Parameters
----------
a : array_like
    Elements to sum.
```

www.ethans.co.in

Slide 278

Sort function

Ethans
Learn from experts

```
# Sort function
>>> a
array([ 2,  4,  5,  1,  2,  3, 20, 10, 15])
>>> a.sort()
>>> a
array([ 1,  2,  2,  3,  4,  5, 10, 15, 20])
>>> a = np.array([2,4,5,1,2,3,20,10,15])
>>> np.sort(a)
array([ 1,  2,  2,  3,  4,  5, 10, 15, 20])
>>> a
array([ 2,  4,  5,  1,  2,  3, 20, 10, 15])
>>> a.argsort()
array([3, 0, 4, 5, 1, 2, 7, 8, 6])
>>> a[a.argsort()]
array([ 1,  2,  2,  3,  4,  5, 10, 15, 20])
>>> a = np.array([10,2,1])
>>> a.argsort()
array([2, 1, 0])
>>> a[a.argsort()]
array([ 1,  2, 10])
```

Slide 279

www.ethans.co.in

Tile function

Ethans
Learn from experts

```
>>> b = np.array([[1, 2], [3, 4]])
>>> np.tile(b, 2)
array([[1, 2, 1, 2],
       [3, 4, 3, 4]])
>>> np.tile(b, (2, 1))
array([[1, 2],
       [3, 4],
       [1, 2],
       [3, 4]])

>>> c = np.array([1,2,3,4])
>>> np.tile(c,(4,1))
array([[1, 2, 3, 4],
       [1, 2, 3, 4],
       [1, 2, 3, 4],
       [1, 2, 3, 4]])
```

Slide 280

www.ethans.co.in