Lezione 3

WebSeminar
Mathematica

Gestione dati con *Mathematica* : Import / Export

Crescenzio Gallo – Università di Foggia
crescenzio.gallo@unifg.it

Note:

» Il materiale visualizzato durante questo seminario è disponibile per il download all'indirizzo http://www.crescenziogallo.it/unifg/seminario-mathematica-2014/

- Il materiale utilizzato è tratto dai webinar pubblicati da Adalta e prodotti dal dott. Roberto Cavaliere (*Mathematica* Technical Sales Manager, *r.cavaliere@adalta.it*)

---

12 – 26 Giugno 2014

Agenda

Introduzione

- *Mathematica* è un linguaggio di sviluppo
- Differenti modi di eseguire *Mathematica*

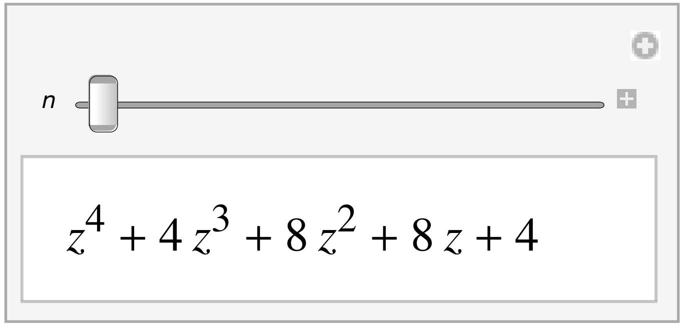
Integrazione/interazione con altri ambienti

- Possibili link

Introduzione: *Mathematica* è un linguaggio di sviluppo

Poichè siamo abituati ad usare *Mathematica* attraverso il front end capita spesso di pensare di non considerare la vera natura di *Mathematica* ossia quella di linguaggio di programmazione.
Già nel fare un semplice calcolo e renderlo parametrico e farlo computare al variare del parametro in effetti abbiamo sfruttato la capacità di programmazione del linguaggio *Mathematica*.

```
Manipulate[Expand[((z + 1)^2 + 1)^n], {n, 2, 10, 1}]
```



$z^4 + 4 z^3 + 8 z^2 + 8 z + 4$

Nonostante *Mathematica* sia un linguaggio funzionale, di fatto mette a disposizione set di funzioni dedicati ai vari approcci di programmazione:

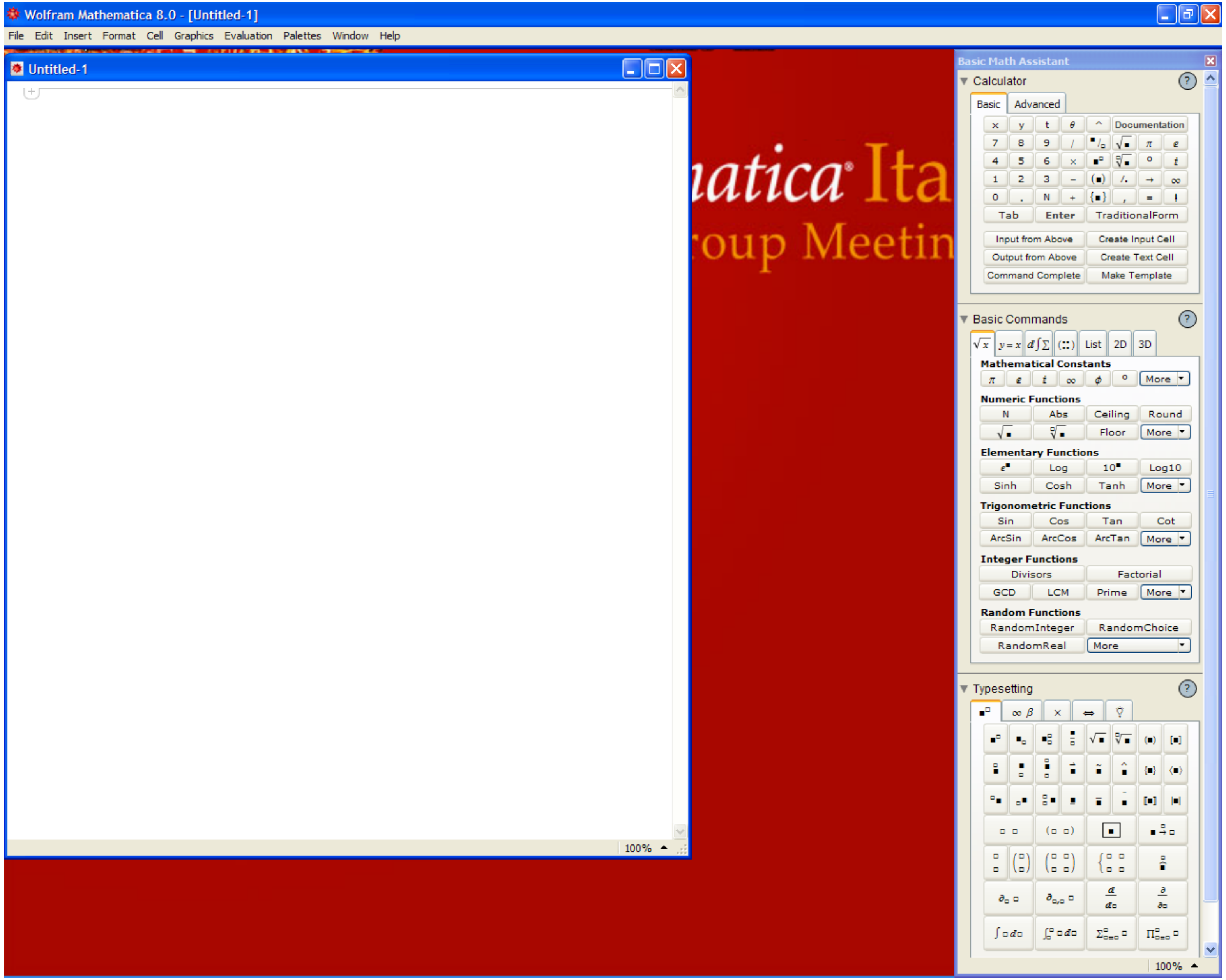
Functional Programming
Procedural Programming
Rule-Based Programming
Graph Programming
CUDA Programming

Dunque un’altra delle sua caratteristiche è che si presenta come un linguaggio **pluriparadigmatico**.

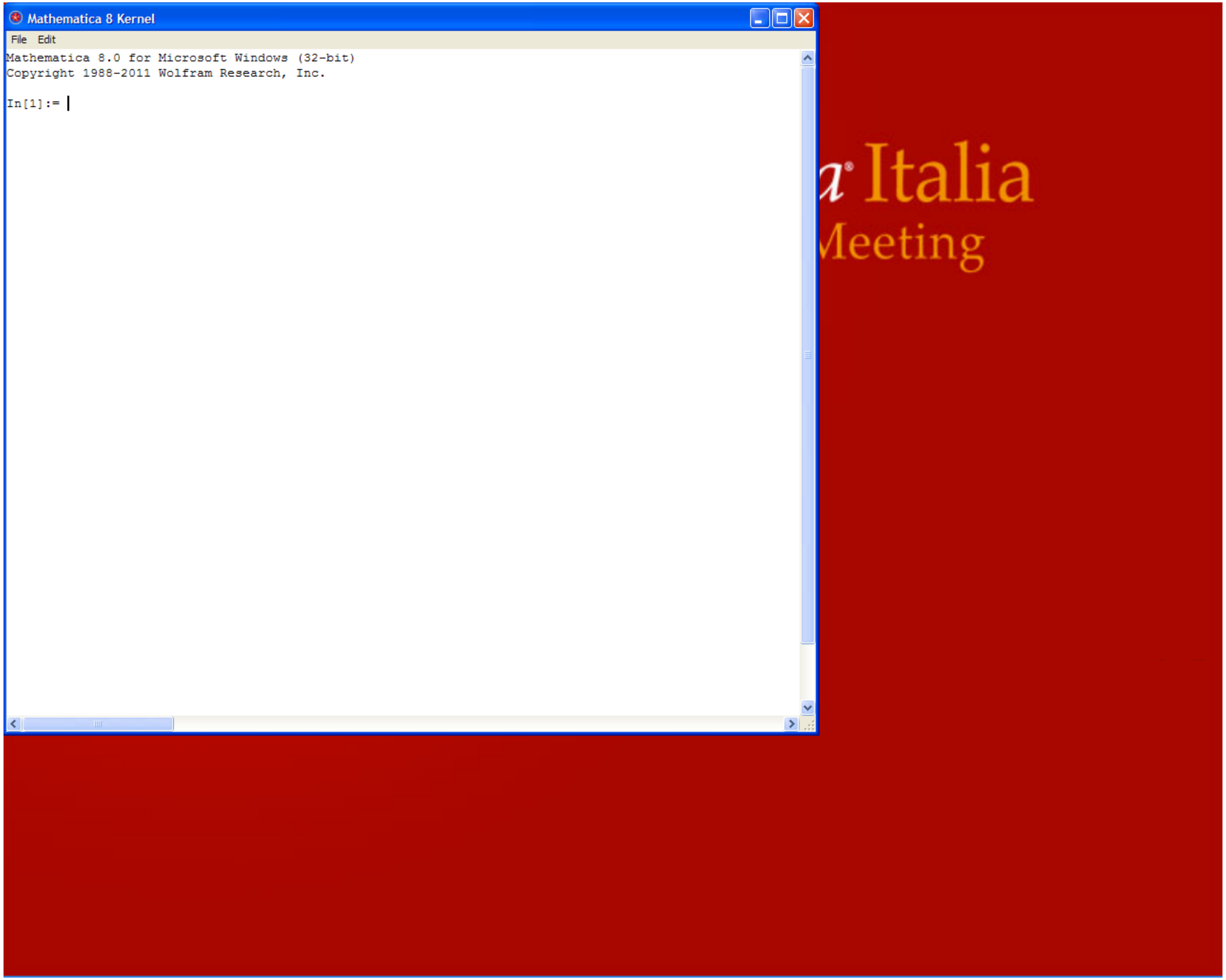
Introduzione: differenti modi di eseguire *Mathematica*

L’architettura interna di *Mathematica* consiste di due moduli separati:

L’interfaccia utente o **front end**: il modulo dedicato al dialogo con l’utente (interpretazioni delle linee di input e rappresentazione degli output).



Il motore di calcolo o **kernel**, dedicato alle computazioni vere e proprie



```
Mathematica 8.0 for Microsoft Windows (32-bit)
Copyright 1988-2011 Wolfram Research, Inc.

In[1]:=
```

I due moduli sono in collegamento attraverso il *MathLink*, una libreria in grado di far dialogare il kernel con diversi altri sistemi. Questa architettura consente l’impiego di *Mathematica* in moltissimi contesti differenti e secondo moltissime modalità.

Ad esempio si può usare il semplice kernel se bisogna fare solo calcoli e non si necessità di controllare “visivamente” l’esecuzione dei calcoli stessi.

◀ | ▶

Integrazione/interazione con altri ambienti: possibili link

L’architettura sopra descritta, ossia la separazione tra il front end ed il kernel ha consentito lo sviluppo di numerose altre interfacce per il kernel di *Mathematica*. Infatti il kernel si può integrare con linguaggio come il C C++, Fortran, Java, .NET ed altri (Systems Interfaces & Deployment).

Vediamo un’applicazione che utilizza Java per l'interfaccia utente ed il kernel di *Mathematica* per i calcoli: `EquationTrekker`

Integrazione/interazione con altri ambienti: Import/Export di dati

Mathematica consente di importare dati in diversi modi.

» Formati standard riconosciuti in import (circa 170 formati)

» Accesso ai database (MySQL, Oracle, ...)

» Dati curati (FinancialData, WeatherData, CountryData, ...)

Integrazione/interazione con altri ambienti: Import/Export di dati

Alcune novità in *Mathematica* 8:

- Geospatial Formats

  ArcGRID, GPX, GRIB, KML/KMZ, NDK, SurferGrid, TLE

- Graph Formats

  DIMACS, DOT, Graphlet, GraphML, GXL, LEDA, Pajek, TGF

- » Multimedia and graphics formats

  FLAC, MKV, QuickTime (export), VideoFrames

- » Miscellaneous formats »

  C, ICS, JSON, NASACDF, VTK, XLSX

```
$ImportFormats
```

```
{3DS, ACO, Affymetrix, AgilentMicroarray, AIFF, ApacheLog, ArcGRID, AU, AVI,
  Base64, BDF, Binary, Bit, BMP, Byte, BYU, BZIP2, CDED, CDF, Character16,
  Character8, CIF, Complex128, Complex256, Complex64, CSV, CUR, DBF,
  DICOM, DIF, DIMACS, Directory, DOT, DXF, EDF, EPS, ExpressionML,
  FASTA, FASTQ, FCS, FITS, FLAC, GenBank, GeoTIFF, GIF, GPX, Graph6,
  Graphlet, GraphML, GRIB, GTOPO30, GXL, GZIP, HarwellBoeing, HDF,
  HDF5, HIN, HTML, ICC, ICNS, ICO, ICS, Integer128, Integer16, Integer24,
  Integer32, Integer64, Integer8, JCAMP−DX, JPEG, JPEG2000, JSON, JVX,
  KML, LaTeX, LEDA, List, LWO, MAT, MathML, MBOX, MDB, MGF,
  MIDI, MMCIF, MOL, MOL2, MPS, MTP, MTX, MX, NASACDF, NB, NDK,
  NetCDF, NEXUS, NOFF, OBJ, ODS, OFF, OpenEXR, Package, Pajek, PBM,
  PCX, PDB, PDF, PGM, PLY, PNG, PNM, PPM, PXR, QuickTime, RawBitmap,
  Real128, Real32, Real64, RIB, RSS, RTF, SCT, SDF, SDTS, SDTSDEM, SFF,
  SHP, SMILES, SND, SP3, Sparse6, STL, String, SurferGrid, SXC, Table, TAR,
  TerminatedString, Text, TGA, TGF, TIFF, TIGER, TLE, TSV, UnsignedInteger128,
  UnsignedInteger16, UnsignedInteger24, UnsignedInteger32, UnsignedInteger64,
  UnsignedInteger8, USGSDEM, UUE, VCF, VCS, VTK, WAV, Wave64, WDX,
  XBM, XHTML, XHTMLMathML, XLS, XLSX, XML, XPORT, XYZ, ZIP}
```

```
$ExportFormats
```

```
{3DS, ACO, AIFF, AU, AVI, Base64, Binary, Bit, BMP, Byte, BYU, BZIP2,
  C, CDF, Character16, Character8, Complex128, Complex256, Complex64,
  CSV, CUR, DICOM, DIF, DIMACS, DOT, DXF, EMF, EPS, ExpressionML,
  FASTA, FASTQ, FCS, FITS, FLAC, FLV, GIF, Graph6, Graphlet, GraphML,
  GXL, GZIP, HarwellBoeing, HDF, HDF5, HTML, ICNS, ICO, Integer128,
  Integer16, Integer24, Integer32, Integer64, Integer8, JPEG, JPEG2000, JSON,
  JVX, KML, LEDA, List, LWO, MAT, MathML, Maya, MGF, MIDI, MOL,
  MOL2, MTX, MX, NASACDF, NB, NetCDF, NEXUS, NOFF, OBJ, OFF,
  Package, Pajek, PBM, PCX, PDB, PDF, PGM, PICT, PLY, PNG, PNM, POV,
  PPM, PXR, QuickTime, RawBitmap, Real128, Real32, Real64, RIB, RTF,
  SCT, SDF, SND, Sparse6, STL, String, SurferGrid, SVG, SWF, Table, TAR,
  TerminatedString, TeX, Text, TGA, TGF, TIFF, TSV, UnsignedInteger128,
  UnsignedInteger16, UnsignedInteger24, UnsignedInteger32, UnsignedInteger64,
  UnsignedInteger8, UUE, VideoFrames, VRML, VTK, WAV, Wave64, WDX,
  X3D, XBM, XHTML, XHTMLMathML, XLS, XLSX, XML, XYZ, ZIP, ZPR}
```

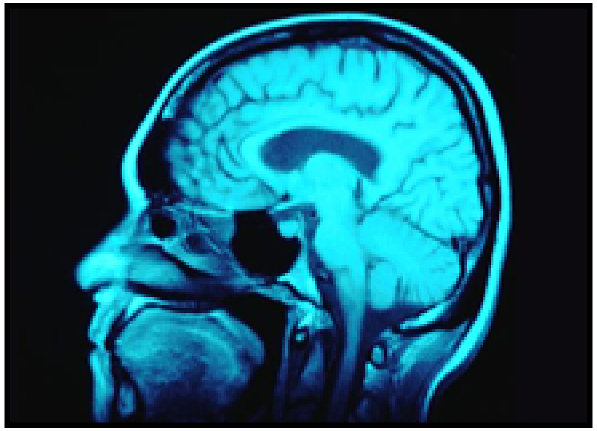
Integrazione/interazione con altri ambienti: Import/Export di dati

Sintassi di base della Import

- `Import["`*file*`",`*format*`]`: importa dati dal file specificato restituendono la versione Mathematica
- `Import["`*file*`", {`*format*`,` *elements*`}]`: importa dal file solo gli elementi specificati.

comunque in grado di identificare il formato automaticamente dal file stesso

```
Import["ExampleData/mri.pxr"]
```



```
Import["ExampleData/cities.xls"]
```

({City, Country, Population}   {Tokyo, Japan, 8.3×10^{6}}   {Chicago, United States, 2.8×1

Se un determinato formato di dati è incluso nella lista **$ImportFormats** significa che *Mathematica* è in grado di gestire anche tutti i metadati associati a tale formato

```
Import[ "http://exampledata.wolfram.com/ArcGRID.zip",
 {"ArcGRID", "Elements"}]
```

```
{Centering, CentralScaleFactor, CoordinateSystem, CoordinateSystemInformation, Data,
  DataFormat, Datum, ElevationRange, Graphics, GridOrigin, Image, InverseFlattening,
  LinearUnits, Projection, ProjectionName, RasterSize, ReferenceModel, ReliefImage,
  SemimajorAxis, SemiminorAxis, SpatialRange, SpatialResolution, StandardParallels}
```

Questo esempio crea una tabella con tutti i metadati contenuti in un file DICOM utilizzato dalle apparecchiature di diagnostica per immagini

```
Import["ExampleData/cities.xls", "Elements"]
```

```
{Data, FormattedData, Formulas, Images, Sheets}
```

TransferSyntaxUID	{1, 2, 840, 10 008, 1, 2, 1}
ImplementationClassUID	{1, 2, 840, 113 654, 2, 3, 1995, 2, 12, 0}
ImplementationVersionName	MIRCTN16NOV2000
SpecificCharacterSet	ISO_IR 100
ImageType	{ORIGINAL, PRIMARY, M_FFE, M, FFE}
InstanceCreationDate	{2005, 3, 22}
InstanceCreationTime	{23, 17, 0}
InstanceCreatorUID	{1, 3, 46, 670 589, 11, 10 453, 5}
SOPClassUID	{1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}
SOPInstanceUID	{1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 223 165 618 714}
StudyDate	{2005, 3, 22}
SeriesDate	{2005, 3, 22}
AcquisitionDate	{2005, 3, 22}
ContentDate	{2005, 3, 22}
StudyTime	{22, 8, 40}
SeriesTime	{22, 13, 34.6}
AcquisitionTime	{22, 13, 34.6}
ContentTime	{22, 13, 34.6}
AccessionNumber	2200000000
Modality	MR
Manufacturer	Philips Medical Systems
InstitutionName	Praxisgemeinschaft Aachen

StationName	PMSN−HYDWC6QW68
StudyDescription	TEST
SeriesDescription	HIRN T1/3D/FFE**
ManufacturerModel	Intera
ReferencedStudyComponentSequence	{SpecificCharacterSet → ISO_IR 100, InstanceCreationDate → {2005, 3, 22}, InstanceCreationTime → {23, 17, 2}, InstanceCreatorUID → {1, 3, 46, 670 589, 11, 10 453, 5}, ReferencedSOPClassUID → {1, 2, 840, 10 008, 3, 1, 2, 3, 3}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 3320, 2 005 032 222 082 071 829}, InstanceNumber → 0}

ReferencedImageSequence	{{ReferencedSOPClassUID → {1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 222 092 884 501}}, {ReferencedSOPClassUID → {1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 222 092 867 498}}, {ReferencedSOPClassUID → {1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 222 092 871 499}}}
(0009,0010)	SECTRA_Ident_01
(0009,1001)	2200000000
(0009,1002)	00
PatientName	{Anonymous, Male 1958}
PatientID	GH−WISE:68736
PatientIDIssuer	anonymous
PatientBirthDate	anonymous
PatientBirthTime	anonymous

PatientSex	M
PatientBirthName	anonymous
PatientMotherBirthName	anonymous
MedicalRecordLocator	anonymous
MedicalAlerts	anonymous
ContrastAllergies	anonymous
ResidenceCountry	anonymous
ResidenceRegion	anonymous
PatientTelephoneNumbers	anonymous
EthnicGroup	anonymous
Occupation	anonymous
SmokingStatus	anonymous
PregnancyStatus	{12 336, 13 360}
PatientReligiousPreference	anonymous
PatientComments	anonymous
ScanningSequence	GR
SequenceVariant	SP
SliceThickness	1.6
RepetitionTime	25.
EchoTime	4.6012
InversionTime	0.
AveragesCount	4.
ImagingFrequency	63.8985

ImagedNucleus	1H
EchoNumbers	1
MagneticFieldStrength	1.5
SlicesSpacing	0.8
PhaseEncodingSteps	416
EchoTrainLength	0
PercentSampling	80.
PercentPhaseFieldOfView	79.8077
DeviceSerialNumber	10453
SoftwareVersions	{NT 10.3.1, PIIM V2.1.4.1 MIMIT MCS}
Protocol	T1/3D/FFE∗∗ SYN+
LowRRValue	0
HighRRValue	0
IntervalsAcquired	0
IntervalsRejected	0
HeartRate	0
ReconstructionDiameter	360.
ReceivingCoil	Syn−head/neck
TransmittingCoil	B
AcquisitionMatrix	{0, 416, 416, 0}
PhaseEncodingDirection	ROW
FlipAngle	30.
PatientPosition	HFS

StudyInstanceUID	{1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 3320, 2 005 032 222 082 070 828}
SeriesInstanceUID	{1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 3616, 2 005 032 222 133 460 760}
StudyID	00
SeriesNumber	2
AcquisitionNumber	3
InstanceNumber	101
ImagePosition	{14.8496, −201.474, 142.196}
ImageOrientation	{−0.0196207, 0.999807, 0., 0., 0., −1.}
ReferenceFrameUID	{1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 3616, 2 005 032 222 084 848 741}
TemporalPositionIdentifier	1
TemporalPositionCount	1
SliceLocation	79.9997
SamplesPerPixel	1
PhotometricInterpretation	MONOCHROME2
Rows	512
Columns	512
PixelSpacing	{0.703125, 0.703125}
PixelAspectRatio	{1, 1}
AllocatedBits	16

StoredBits	12
HighBit	11
PixelRepresentation	0
WindowCenter	994.
WindowWidth	1987.
RescaleIntercept	0.
RescaleSlope	1.53968
RescaleType	normalized
LossyImageCompression	00
StudyComments	TEST
PerformedStationAETitle	INTERA
PerformedProcedureStartDate	{2005, 3, 22}
PerformedProcedureStartTime	{22, 8, 40}
PerformedProcedureEndDate	{2005, 3, 22}
PerformedProcedureEndTime	{22, 8, 40}
PerformedProcedureID	164758100
PerformedProcedureDescription	TEST
RequestAttributesSequence	{}
PerformedProceduresComments	TEST
FilmConsumptionSequence	{}
(2001,0010)	Philips Imaging DD 001
(2001,1001)	0.
(2001,1002)	0

(2001,1003)	0.
(2001,1006)	N
(2001,1007)	U
(2001,1008)	1
(2001,1009)	0.
(2001,100A)	101
(2001,100B)	SAGITTAL
(2001,100C)	N
(2001,100E)	N
(2001,100F)	0
(2001,1010)	NO
(2001,1011)	0.
(2001,1012)	N
(2001,1013)	1
(2001,1014)	1
(2001,1015)	1
(2001,1016)	0
(2001,1017)	1
(2001,1018)	220
(2001,1019)	N
(2001,101A)	{0., 0., 0.}
(2001,101B)	0.
(2001,101C)	NO

(2001,101D)	1
(2001,101F)	NO
(2001,1020)	T1FFE
(2001,1021)	N
(2001,1022)	1.39438
(2001,1023)	30.
(2001,1024)	N
(2001,1025)	4.6
(2001,105F)	{(2001,0010) → Philips Imaging DD 001, (2001,102D) → 220, (2001,1032) → 0., (2001,1033) → AP, (2001,1035) → 1, (2001,1036) → PARALLEL, (2005,0010) → Philips MR Imaging DD 001, (2005,1071) → 0., (2005,1072) → 1.12425, (2005,1073) → 0., (2005,1074) → 287.308, (2005,1075) → 360., (2005,1076) → 176., (2005,1078) → −21.6573, (2005,1079) → −37.8042, (2005,107A) → 3.71936, (2005,107B) → AP, (2005,107E) → 0.8, (2005,1081) → RL, (2005,10A3) → 39, (2005,10A4) → 7, (2005,10A5) → 0, (2005,10A6) → 123, (2005,10A7) → DEFAULT}

(2001,1060)	1
(2001,1061)	N
(2001,1062)	N
(2001,1063)	ELSEWHERE
(2001,106E)	MRSERIES
(2001,107B)	3
(2001,1081)	1
(2001,1082)	0
(2001,1083)	63.8985
(2001,1084)	0.
(2001,1085)	0.
(2001,1086)	0
(2001,1087)	1H
(2001,1088)	4.
(2001,1089)	0.
(2001,108A)	0.
(2001,108B)	B
(2005,0010)	Philips MR Imaging DD 001
(2005,0011)	Philips MR Imaging DD 002
(2005,0012)	Philips MR Imaging DD 003
(2005,0013)	Philips MR Imaging DD 004
(2005,1000)	0.
(2005,1001)	1.12425

(2005,1002)	0.
(2005,1004)	NONE
(2005,1008)	−21.5082
(2005,1009)	−37.8042
(2005,100A)	11.3179
(2005,100B)	1.41824×10^6
(2005,100C)	0.212523
(2005,100D)	0.
(2005,100E)	0.00121813
(2005,1011)	M
(2005,1012)	N
(2005,1013)	NO
(2005,1014)	N
(2005,1015)	N
(2005,1016)	N
(2005,1017)	Y
(2005,1019)	N
(2005,101A)	0
(2005,101B)	N
(2005,101C)	N
(2005,101D)	416
(2005,101E)	compose

(2005,101F)	compose
(2005,1020)	0
(2005,1021)	1
(2005,1022)	0
(2005,1023)	0
(2005,1025)	0
(2005,1026)	N
(2005,1027)	MAXIMUM
(2005,1028)	N
(2005,1029)	N
(2005,102A)	170 719 738
(2005,102B)	100
(2005,102C)	N
(2005,102D)	3
(2005,102E)	N
(2005,102F)	N
(2005,1030)	25.
(2005,1031)	N
(2005,1033)	3737.5
(2005,1034)	N
(2005,1035)	PIXEL
(2005,1036)	N
(2005,1037)	N

(2005,1038)	Y
(2005,1039)	Y
(2005,103A)	HIRN
(2005,103B)	N
(2005,103C)	N
(2005,103D)	102
(2005,103E)	{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
(2005,105F)	UNKNOWN
(2005,1060)	−1
(2005,1061)	NO
(2005,1063)	0
(2005,106E)	FFE
(2005,106F)	3D

(2005,1080)	{{ReferencedSOPClassUID → {1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 222 092 884 501}, (2005,0010) → Philips MR Imaging DD 001, (2005,103F) → 1, (2005,1040) → 0, (2005,1041) → 1, (2005,1042) → 1, (2005,1043) → M, (2005,1044) → 1, (2005,1045) → 1, (2005,1046) → FFE, (2005,1047) → 2}, {ReferencedSOPClassUID → {1, 2, 840, 10 008, 5, 1, 4, 1, 1, 4}, ReferencedSOPInstanceUID → {1, 3, 46, 670 589, 11, 0, 0, 11, 4, 2, 0, 10 453, 5, 4100, 2 005 032 222 092 867 498}, (2005,0010) → Philips MR Imaging DD 001, (2005,103F) → 1, (2005,1040) → 0, (2005,1041) → 1, (2005,1042) → 1, (2005,1043) → M, (2005,1044) → 1, (2005,1045) → 1, (2005,1046) → FFE, (2005,1047) → 4}, {ReferencedSOPClassUID →

(2005,1083)	{}
(2005,1084)	{}
(2005,1085)	{}
(2005,1086)	0
(2005,109E)	{}
(2005,10A0)	0.
(2005,10A1)	SYN_COCA
(2005,10A2)	N
(2005,10A8)	0.
(2005,10B0)	0.
(2005,10B1)	0.
(2005,10B2)	0.
(2005,10C0)	GR
(2005,1199)	0
(2005,1200)	1
(2005,1201)	0
(2005,1243)	0
(2005,1244)	0
(2005,1245)	1
(2005,1246)	0
(2005,1247)	0
(2005,1248)	0
(2005,1249)	0

(2005,1250)	0
(2005,1251)	0
(2005,1252)	0
(2005,1253)	0
(2005,1254)	0
(2005,1255)	0
(2005,1325)	N
(2005,1326)	0.
(2005,1327)	REAL
(2005,1328)	ORIGINAL
(2005,1329)	50.
(2005,1330)	{PLUS_A_PLUS_B, PLUS_A_PLUS_B, PLUS_A_PLUS_B, PLUS_A_PLUS_B}
(2005,1331)	0
(2005,1333)	{0., 0., 0.}
(2005,1334)	UNKNOWN
(2005,1335)	UNKNOWN
(2005,1336)	0.
(2005,1337)	0.
(2005,1338)	0.
(2005,1339)	{0, 0}
(2005,1340)	PRE_FT
(2005,1341)	UNKNOWN

(2005,1342)	FID
(2005,1343)	Y
(2005,1344)	{0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}
(2005,1345)	NO
(2005,1346)	HERTZ
(2005,1347)	0.
(2005,1348)	OFF
(2005,1349)	0.
(2005,1350)	{0., 0.}
(2005,1351)	0
(2005,1352)	0
(2005,1355)	{0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.}
(2005,1356)	NO
(2005,1357)	0
(2005,1359)	1.
(2005,1360)	0.
(2005,1361)	{0., 0.}
(2005,1362)	0.
(2005,1363)	0.

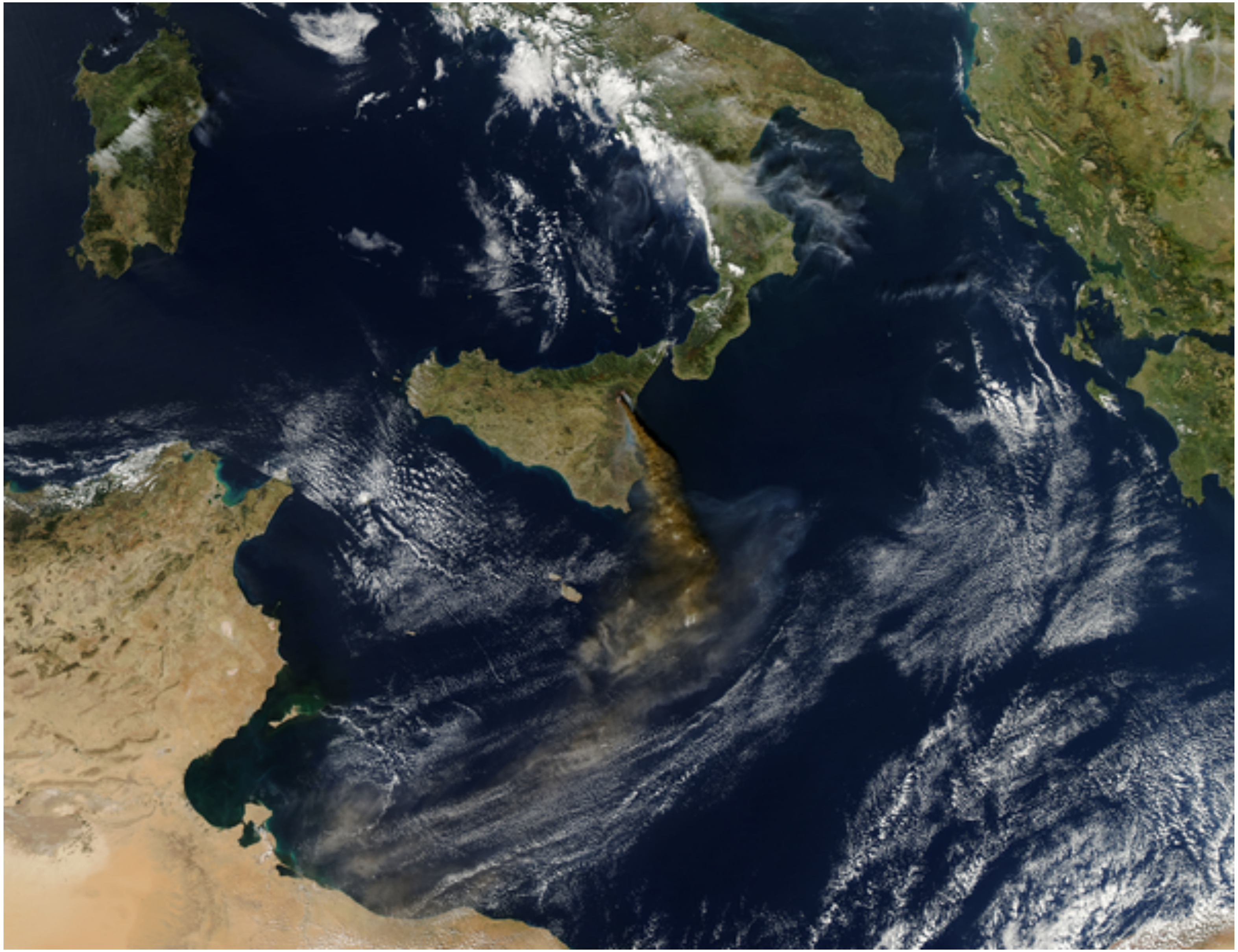
(2005,1364)	NO
(2005,1370)	0
(2005,1371)	{}
(2005,1381)	3

◀ | ▶

Integrazione/interazione con altri ambienti: Import/Export di dati

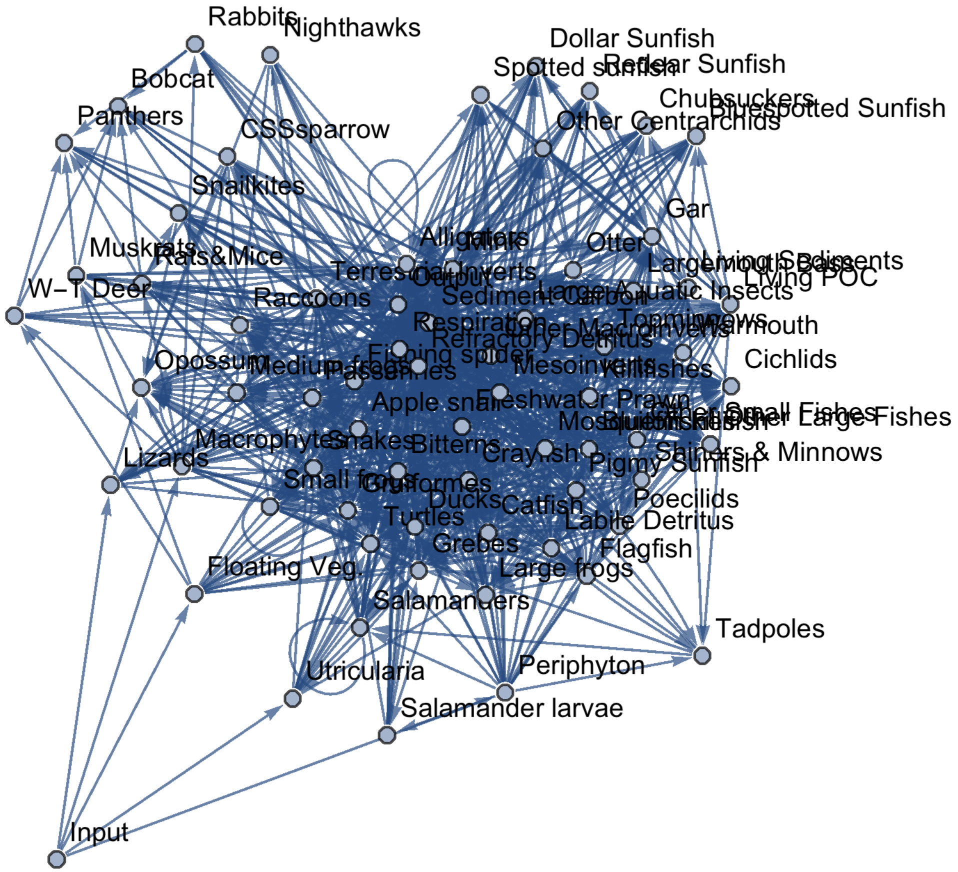
Import è in grado di importare sia dal file system su cui è installato *Mathematica* sia da reti intranet/Internet, semplicemenete attraverso l’URL.

Questa è l’immagine dell’eruzione dell’Etna dell’Ottobre 2002 vista dal satellite Aqua



Import di un formato Graph (Pajek)

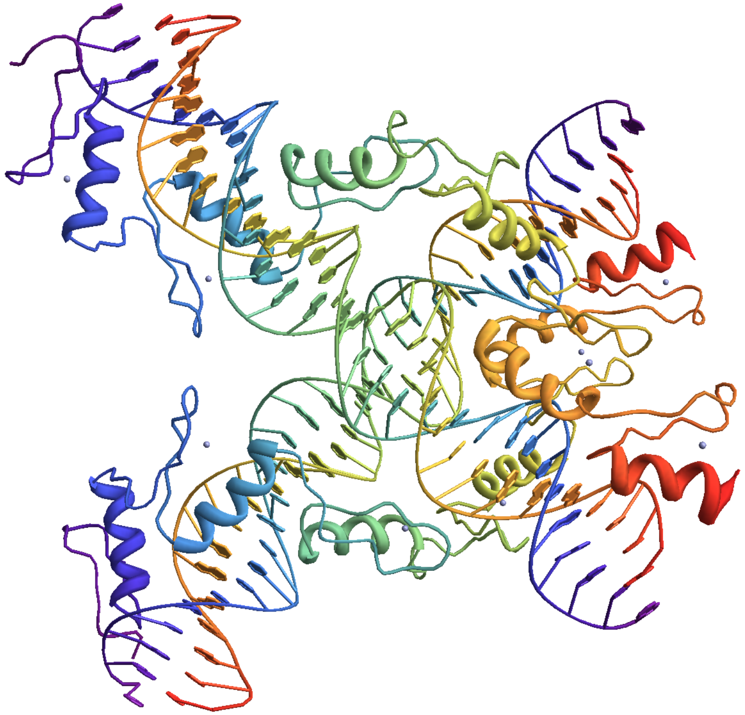
```
Import[
 "http://vlado.fmf.uni-lj.si/pub/networks/data/bio/foodweb/
   Everglades.paj", "Pajek"]
```



Import di un file JPG

Una molecola in formato PDB

```
Import["http://www.rcsb.org/pdb/download/downloadFile.do?
   fileFormat=pdb&compression=NO&structureId=1tf6",
 "PDB", ImageSize → Medium]
```



Integrazione/interazione con altri ambienti: Import/Export di dati

Vediamo qualche esempio di import di dati oltre che di immagini.

Importare direttamente da pagine HTML

Importiamo la pagina BLS web page

```
Import["http://www.bls.gov/web/laus/laumstrk.htm"]
```

```
Skip to Content


 US Department of Labor
 A to Z Index |  FAQs |  About BLS |  Contact Us  E–mail:
```

U.S. Bureau of Labor Statistics

Follow Us | What's New | Release Calendar | Site Map

Home

Subject Areas» Inflation & Prices Spending & Time Use Unemployment Employment Pay & Benefits Productivity Workplace Injuries International Demographics Industries Business Costs Occupations Geography
Resources For » Business Leaders Consumers Developers Economists Investors Jobseekers Media Public Policymakers Students & Teachers Survey Respondents At a Glance Tables » U.S. Economy Regions, States, & Areas Industries International
MORE » Announcements Commissioner's Corner Green Jobs Research Spotlight on Statistics Students' Pages Errata Other Statistical Sites Subjects

Inflation & Prices » Consumer Price Index Producer Price Indexes Import/Export Price Indexes Employment Cost Index Contract Escalation Price Index

```
    Research Inflation & Prices Overview   Pay & Benefits »   Employment Costs
    National Compensation Data Wages by Area & Occupation Earnings by
    Demographics Earnings by Industry County Wages Benefits Compensation
    Research Strikes & Lockouts Pay & Benefits Overview   Spending &
    Time Use »   Consumer Expenditures How Americans Spend Time
Unemployment »   National Unemployment Rate State & Local Unemployment
    Rates Mass Layoffs Unemployment Research Unemployment Overview
    Employment »   National Employment State & Local Employment State &
    County Employment Worker Characteristics Employment Projections Job
    Openings & Labor Turnover Green Goods and Services Green Goods and
    Services Occupations Green Technologies and Practices Employment by
    Occupation Work Experience Over Time Business Employment Dynamics
    Employment Research Employment Overview   Workplace Injuries »
Productivity »   Labor Productivity & Costs Multifactor Productivity Productivity
    Research Productivity Overview   International »   International Labor
    Comparisons International Technical Cooperation Import/Export Price
    Indexes International Overview   REGIONAL OFFICES »   New England
    (Boston) New York−New Jersey (NY City) Mid−Atlantic (Philadelphia)
    Southeast (Atlanta) Midwest (Chicago) Southwest (Dallas) Mountain−Plains
    (Kansas City) West (San Francisco) Regional Overview    Data Tools

Data Retrieval Tools »   Top Picks Series Report
    One Screen Multi−Screen Maps Calculators Public Data API
```

CUSTOMIZED TABLES » Text Files News Release Tables
More Sources of Data » Discontinued Databases FAQs
Special Notices More Sources of Data Publications

Latest Publications » The Editor's Desk Monthly Labor Review Beyond
the Numbers National Longitudinal Surveys News Spotlight on Statistics
Occupational Outlook Handbook Occupational Outlook Quarterly
Magazines & Journals Chartbooks Bulletins & Reports Catalog of Publications
Research Papers Copyright Information Contact & Help Economic Releases

Latest Releases » Major Economic Indicators » Schedules for news Releases » By
Month By News Release Current Year Prior Years Archived News Releases »
Employment & Unemployment » Monthly Quarterly, annual, and
other Inflation & Prices » Pay & Benefits & Workplace Injuries
» Productivity & Technology » Employment Projections »
International Programs » Regional News Releases » Students

Games & Quizzes Student Resources Teacher´s Desk History of BLS FAQs Beta

Data API » Data Finder » State and County Map »

Local Area Unemployment Statistics

FONT SIZE:
LAU
Local Area Unemployment Statistics
LAU Homepage LAU Overview LAU FAQ Contact LAU
SHARE ON:

BROWSE LAU LAU HOME LAU OVERVIEW LAU
NEWS RELEASES LAU DATABASES LAU TABLES & MAPS
LAU DOCUMENTATION LAU FAQs CONTACT LAU
SEARCH LAU

LAU TOPICS JOBSEEKERS
INFORMATION ON APPLYING FOR
UNEMPLOYMENT BENEFITS PUBLIC POLICYMAKERS
ADMINISTRATIVE USES OF LOCAL AREA
UNEMPLOYMENT STATISTICS RESEARCHERS
SUBNATIONAL LABOR FORCE ESTIMATION
METHODOLOGY LABOR FORCE DATA

```
DATA AVAILABILITY GEOGRAPHY
GEOGRAPHIC CONCEPTS METHODOLOGY
ESTIMATION METHODOLOGY
  Unemployment Rates for States   Unemployment Rates for States
Monthly Rankings
Seasonally Adjusted
Apr. 2014 p

Rank State Rate
1 NORTH DAKOTA 2.6
2 VERMONT 3.3
3 NEBRASKA 3.6
4 WYOMING 3.7
5 SOUTH DAKOTA 3.8
5 UTAH 3.8
7 IOWA 4.3
8 HAWAII 4.4
8 NEW HAMPSHIRE 4.4
10 LOUISIANA 4.5
11 OKLAHOMA 4.6
12 MINNESOTA 4.7
13 KANSAS 4.8
13 MONTANA 4.8
```

```
15 VIRGINIA 4.9
16 IDAHO 5.0
17 TEXAS 5.2
18 SOUTH CAROLINA 5.3
19 MARYLAND 5.5
20 INDIANA 5.7
20 MAINE 5.7
20 OHIO 5.7
20 PENNSYLVANIA 5.7
24 DELAWARE 5.8
24 WISCONSIN 5.8
26 COLORADO 6.0
26 MASSACHUSETTS 6.0
26 WEST VIRGINIA 6.0
29 WASHINGTON 6.1
30 FLORIDA 6.2
30 NORTH CAROLINA 6.2
32 TENNESSEE 6.3
33 ALASKA 6.4
34 ARKANSAS 6.6
34 MISSOURI 6.6
36 NEW YORK 6.7
37 NEW MEXICO 6.8
```

```
38 ALABAMA 6.9
38 ARIZONA 6.9
38 CONNECTICUT 6.9
38 NEW JERSEY 6.9
38 OREGON 6.9
43 GEORGIA 7.0
44 MICHIGAN 7.4
45 DISTRICT OF COLUMBIA 7.5
45 MISSISSIPPI 7.5
47 KENTUCKY 7.7
48 CALIFORNIA 7.8
49 ILLINOIS 7.9
50 NEVADA 8.0
51 RHODE ISLAND 8.3
  p = preliminary.
 NOTE: Rates shown are a percentage of
      the labor force. Data refer to place of residence. Estimates for
      the current month are subject to revision the following month.


Last Modified Date: May 16, 2014

Recommend this page using:
```

Facebook
Twitter
LinkedIn

tools Areas at a Glance Industries at a Glance Economic Releases Databases & Tables Maps
calculators Inflation Location Quotient Injury And Illness
help Help & Tutorials FAQs Glossary About BLS Contact Us
info What's New Careers @ BLS Find It! DOL Join our Mailing Lists Linking & Copyright Info
resources Inspector General (OIG) Budget and Performance No Fear Act USA.gov Benefits.gov Disability.gov

Freedom of Information Act | Privacy & Security Statement | Disclaimers | Customer Survey | Important Web Site Notices

U.S. Bureau of Labor Statistics | Local Area

```
      Unemployment Statistics Information and Analysis, PSB Suite
      4675, 2 Massachusetts Avenue, NE Washington, DC 20212−0001
    www.bls.gov/LAU | Telephone: 1−202−691−6392 | Contact LAUS
```

Ovviamente risulta poco leggibile, ma se ci interessano i dati contenuti nella pagina possiamo scrivere qualche riga di codice che ci consente di estrarre i soli dati numerici che ci interessano e convertirli facilmente in una tabella *Mathematica*:

VERMONT	3.3
NEBRASKA	3.6
WYOMING	3.7
SOUTH DAKOTA	3.8
UTAH	3.8
IOWA	4.3
HAWAII	4.4
NEW HAMPSHIRE	4.4
LOUISIANA	4.5
OKLAHOMA	4.6
MINNESOTA	4.7
KANSAS	4.8
MONTANA	4.8
VIRGINIA	4.9
IDAHO	5.
TEXAS	5.2
SOUTH CAROLINA	5.3
MARYLAND	5.5
INDIANA	5.7
MAINE	5.7
OHIO	5.7
PENNSYLVANIA	5.7

DELAWARE	5.8
WISCONSIN	5.8
COLORADO	6.
MASSACHUSETTS	6.
WEST VIRGINIA	6.
WASHINGTON	6.1
FLORIDA	6.2
NORTH CAROLINA	6.2
TENNESSEE	6.3
ALASKA	6.4
ARKANSAS	6.6
MISSOURI	6.6
NEW YORK	6.7
NEW MEXICO	6.8
ALABAMA	6.9
ARIZONA	6.9
CONNECTICUT	6.9
NEW JERSEY	6.9
OREGON	6.9
GEORGIA	7.
MICHIGAN	7.4
DISTRICT OF COLUMBIA	7.5

MISSISSIPPI	7.5
KENTUCKY	7.7
CALIFORNIA	7.8
ILLINOIS	7.9
NEVADA	8.
RHODE ISLAND	8.3

Volendo possiamo anche aggiungere l’intestazione di colonna interattiva:

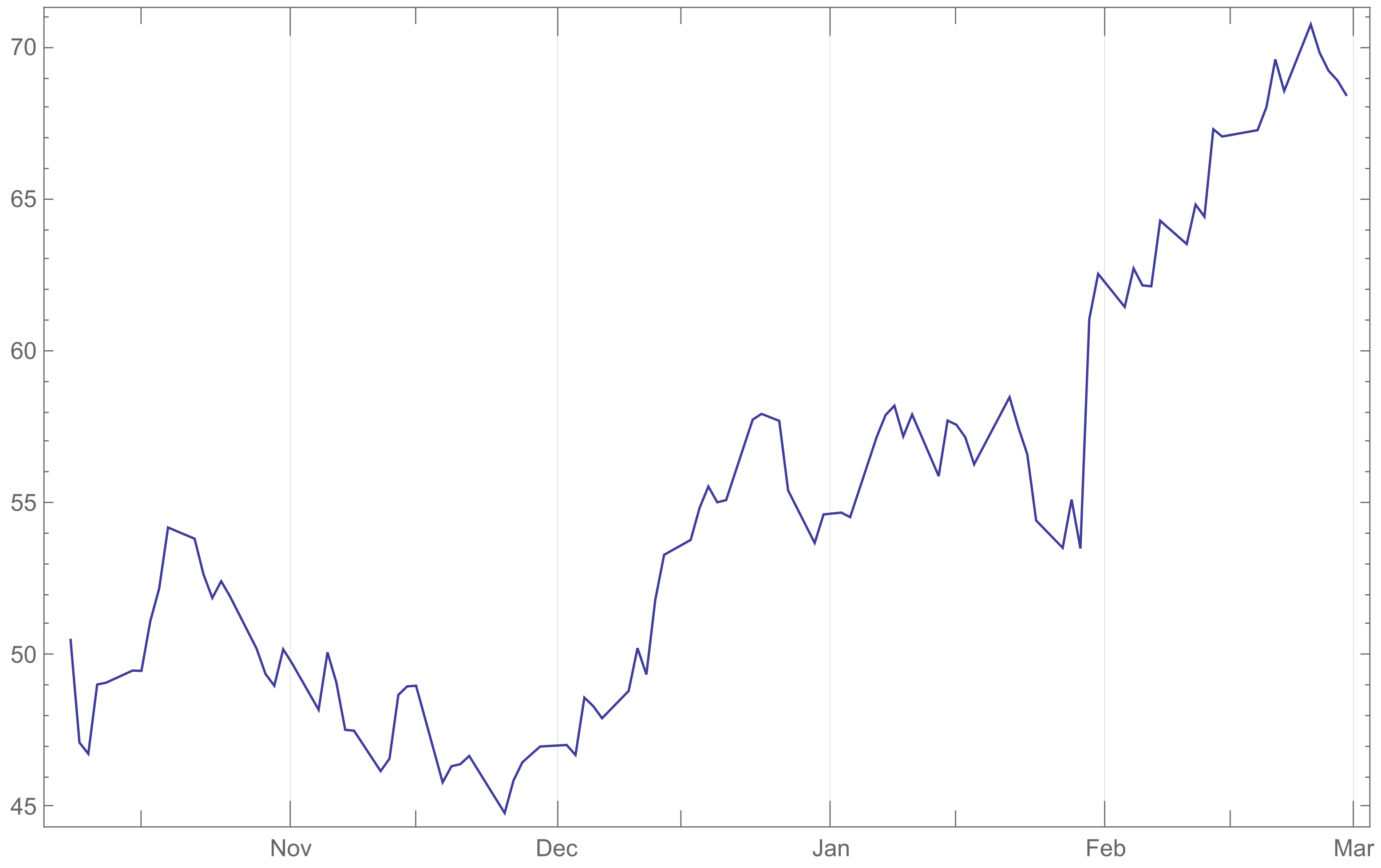
```
Grid[
 {{Grid[
    {{Button[Style["State", {Blue, "Link"}],
       results = SortBy[results, First], Appearance → None],
      Button[Style["Rate", {Blue, "Link"}],
       results = SortBy[results, Last], Appearance → None]}},
    Alignment → {{Left, Right}}, ItemSize → 11]},
  {Dynamic[Grid[results, Alignment → {{Left, Right}},
      Frame → All, ItemSize → 14]]}}, Frame → True,
 Background → LightYellow, BaseStyle → {"Panel"}]
```

Import dati finanziari.

Questo è un esempio di Import di dati dal sito http://finance.yahoo.com

```
facebook =
  Import[
   "http://ichart.finance.yahoo.com/table.csv?s=FB&d=2&e=2&f
     =2014&g=d&a=4&b=18&c=2012&ignore=.csv"];
```

```
DateListPlot[Part[Rest[facebook], 1 ;; 100, {1, 5}],
 Joined → True, ImageSize → Large]
```



```
$ExportFormats
```

```
{3DS, ACO, AIFF, AU, AVI, Base64, Binary, Bit, BMP, Byte, BYU, BZIP2,
  C, CDF, Character16, Character8, Complex128, Complex256, Complex64,
  CSV, CUR, DICOM, DIF, DIMACS, DOT, DXF, EMF, EPS, ExpressionML,
  FASTA, FASTQ, FCS, FITS, FLAC, FLV, GIF, Graph6, Graphlet, GraphML,
  GXL, GZIP, HarwellBoeing, HDF, HDF5, HTML, ICNS, ICO, Integer128,
  Integer16, Integer24, Integer32, Integer64, Integer8, JPEG, JPEG2000, JSON,
  JVX, KML, LEDA, List, LWO, MAT, MathML, Maya, MGF, MIDI, MOL,
  MOL2, MTX, MX, NASACDF, NB, NetCDF, NEXUS, NOFF, OBJ, OFF,
  Package, Pajek, PBM, PCX, PDB, PDF, PGM, PICT, PLY, PNG, PNM, POV,
  PPM, PXR, QuickTime, RawBitmap, Real128, Real32, Real64, RIB, RTF,
  SCT, SDF, SND, Sparse6, STL, String, SurferGrid, SVG, SWF, Table, TAR,
  TerminatedString, TeX, Text, TGA, TGF, TIFF, TSV, UnsignedInteger128,
  UnsignedInteger16, UnsignedInteger24, UnsignedInteger32, UnsignedInteger64,
  UnsignedInteger8, UUE, VideoFrames, VRML, VTK, WAV, Wave64, WDX,
  X3D, XBM, XHTML, XHTMLMathML, XLS, XLSX, XML, XYZ, ZIP, ZPR}
```

◀ | ▶

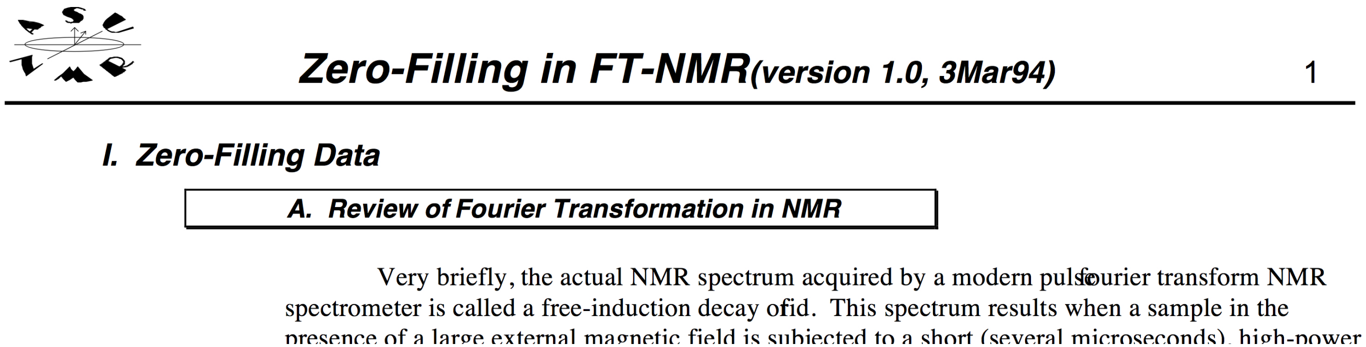
Integrazione/interazione con altri ambienti: Import/Export di dati

Un ultimo esempio di import documenti: il riconoscimento del testo.

```
SetDirectory[NotebookDirectory[]]
```

/Users/cgallo/Box
    Sync/Documenti/Universita/UNIFG/Didattica/Seminario Mathematica 2014

```
img = Import["zerofill.jpg"]
```



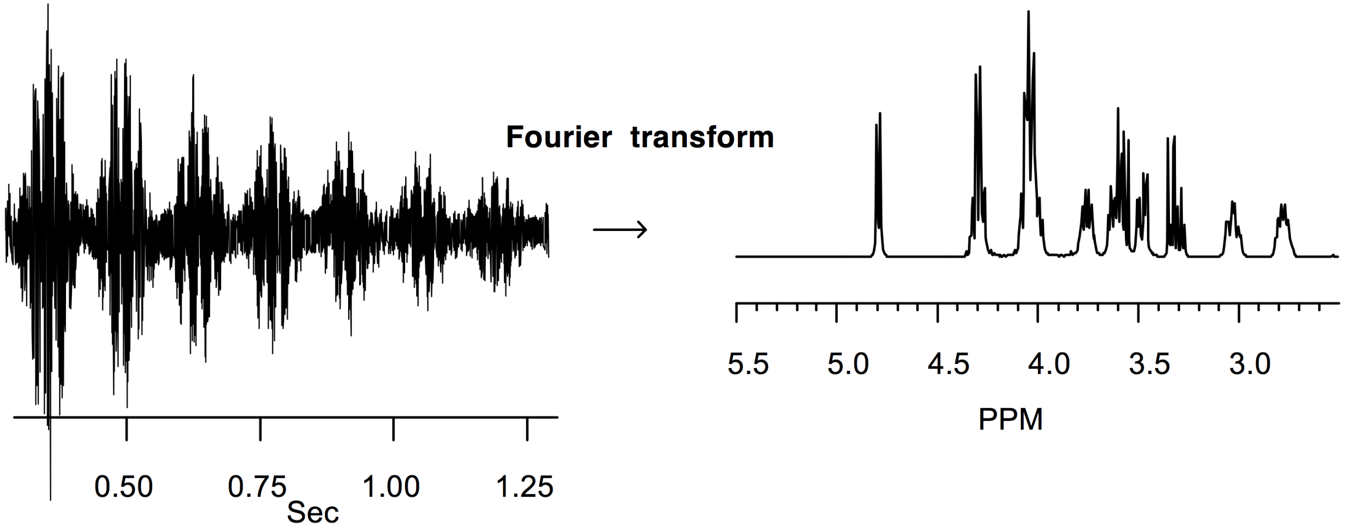
Zero-Filling in FT-NMR *(version 1.0, 3Mar94)* 1

I. Zero-Filling Data

A. Review of Fourier Transformation in NMR

Very briefly, the actual NMR spectrum acquired by a modern pulsefourier transform NMR spectrometer is called a free-induction decay ofid. This spectrum results when a sample in the presence of a large external magnetic field is subjected to a short (several microseconds), high-power presence of a large external magnetic field is subjected to a short (several microseconds), high-power pulse (50-1,000W) of radio-frequency energy at the resonance frequency of the nuclei of interest. This burst of energy is released by the sample over a much longer period of time (typically seconds) as the nuclear spins return to their equilibrium energy states. The released energy is emitted as a radio wave. The frequency of this wave is dependent upon the local magnetic environment of the nuclei. If the excited nuclei in a sample are all in the same magnetic environment, the observed signal will consist of a single decaying radio frequency (sine wave). If there are several magnetically inequivalent nuclei, each will release its absorbed energy at a slightly different frequency. The observed signal will consist of a decayingwaveform which is the sum of the individual decaying sine waves from each of the inequivalent nuclei. This signal *induces* a current in the nmr probe and the signal *decays* as the nuclei freely release their absorbed energy, hence the term free-induction decay.

The actual spectral data acquired by the NMR is the free-induction decay, or **fid**. The "spectrum" we always plot and interpret results from a mathematical manipulation (ft) of the acquired spectral data. In pulse ft-NMR, the fid is fourier transformed. The ft converts the AMPLITUDE vs. TIME domain information in the fid to the AMPLITUDE vs. FREQUENCY domain seen in the typical nmr "spectrum".



The continuous NMR radio-frequency signal (fid) emitted by the sample (for ^{1}H on the Gemini-300, the NMR signals occur at 300MHz, +/-several kilohertz) is reduced to the audio-frequency range by mixing out the high-frequency component. The low-frequency audio spectrum containing the NMR signals is then converted into a discrete series of data points by an analog-to-digital converter. The number of points that you acquire can be controlled with the parameter **np**. Alternatively, the acquisition time **at** can also be used to set the number of points. At a given sweep width, a longer value for **at** will always result in a greater number of data points **np**.

Zero-Filling Ft of Data

```
TextRecognize[img]
```

```
ev
wine Zem−Filling  FT−NMH(versien  aMar94} 1
I; ZEIO−Filling Data
A.. Review ofFou1−fer Transformation in NMR |
  briefly, the actual NMR       la modern pt11®111‘ieret1‘a11sf01‘n1
spectrometer is   a neeanductie  decay nfid.     results whe  as ple   the
preseneeef a large external magnetic field
      'is subjected te a short (several microseconds), highepower
pulse (50−1   of radio−frequency energy at the resonance
      frequency of the nuclei of interest. This
  ef energy is released by the ssmple over a much longer   of time   sece ds) as the
nuclear spins return to their   energy
      states. The released energy is emitted as a radio wave..
The frequency of this wave is dependent upon the local
      magnetic enviroment of the nuclei.   the
excited nuclei in a sample are   inthe same magnetic
      enviromnent, the observed si al   censist
of a single decaying radio frequeney (sine wave),
```

there are several magnetieallpequivalent
nuelei, each Will releaseits abso%ed energy ata slightly
different frequency. The Observed signal will
consist nf an deeayingwavefonn whieh is the ofthe decaying sine waves each of
the 'inequivalent nuclei; This signalinducesea
−eurrent in the n probe and the lsignaldecays as the
nuelei freely release their ahsowed energy, hence the term free−induction. decay.
The~aet11a1.speCtra;1.data by the N is the freeanduetion decay, The
d“speetrum" we always plot and interpret
results from fa rnathernatieal manipulation (ft) ef the acquired
rspeetral pulse ft−NMR., thetid .isfeurier The ft converts the
E infonnation in to vs∗ seen the
‘lspeenurni
Feurier transform
é
|| |1||||1||||||||.|||
PPM
I I I I
0 I 1.06 1

The continuous radio,−frequency signalftd) emitted the sample €fo11H onthe
Ge ini−300, the ;sig a1s−occur at
SDOMHZ, +/eseveral ki1ohertz)is reduced tothe audie−

frequeney range by itnixmgfout the high−frequency
component. The Iowefrequeney audio spectrum
containing the NMR.   is then converted into a discrete
series of data points   an analog−to~
converters... The number ofpoints you acquire can   eontrolled the
A1ternative1y,,.the.acquisition.timeat can
also he used to set the number ofpoints. At, a given sweep
a lenger value forat   always result ina greater number of   pointrnp.

Zero~.FilIiing

Ft of Data

◀ | ▶

Integrazione/interazione con altri ambienti: Import/Export di dati

Come esportare verso altri formati.

Ci sono due modi di esportare dati, immagini o documenti da *Mathematica*:

- **Save As** (menu File o menu contestuale)
- **Export**

Save As è più rapido ma non consente di personalizzare il risultato ai livelli consentiti da **Export** tramite le opzioni. Comunque anche **SaveAs** fornisce una serie di opzioni di base per il salvataggio dei file.

Integrazione/interazione con altri ambienti:

Import/Export di dati

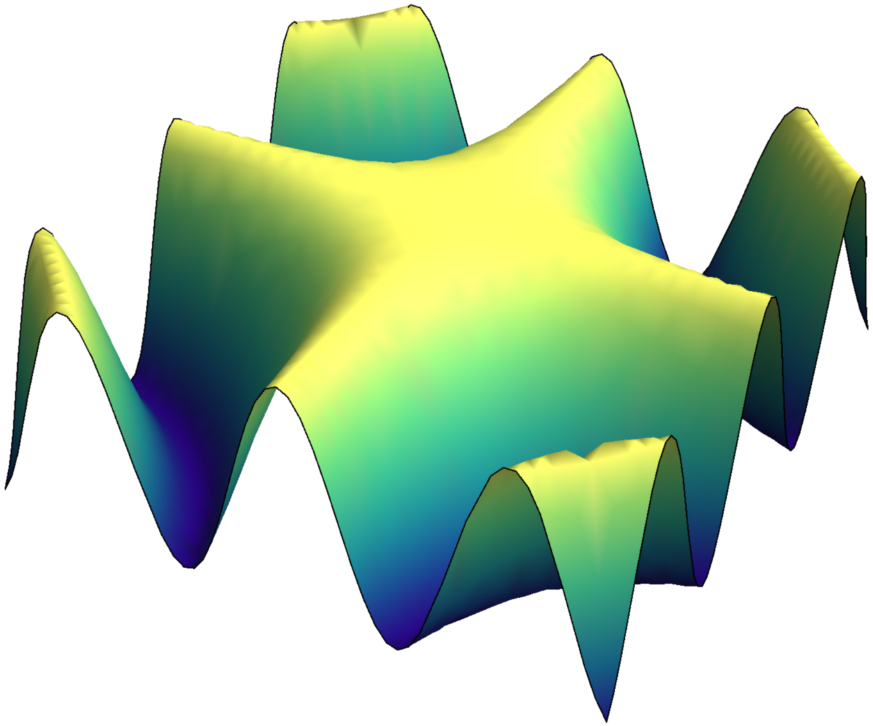
» Export

```
gr = Plot3D[Cos[x y], {x, -3, 3}, {y, -3, 3},
  ColorFunction → "BlueGreenYellow", Mesh → None,
  Boxed → False, Axes → False]
```

Integrazione/interazione con altri ambienti: Import/Export di dati

» Export

```
gr = Plot3D[Cos[x y], {x, -3, 3}, {y, -3, 3},
  ColorFunction → "BlueGreenYellow", Mesh → None,
  Boxed → False, Axes → False]
```



```
Export["Plot3D.jpg", gr, "ColorSpace" → "GrayLevel"]
```

Plot3D.jpg

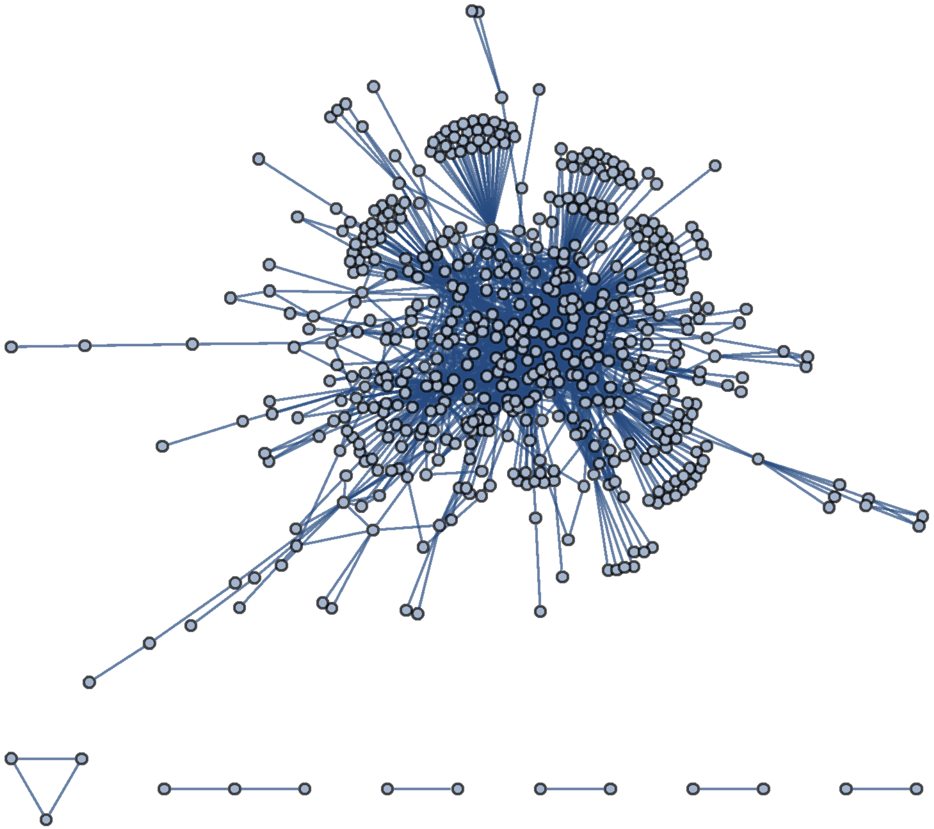
```
an = Manipulate[
  Plot[{BesselJ[a, x], BesselJ[2 a, x], BesselJ[3 a, x],
    BesselJ[4 a, x]}, {x, 0, 10}, Filling → Automatic,
   PlotRange → 1, ImageSize → Large], {a, 0, 2}]
```

```
SystemOpen["BesselJCurves.avi"]
```

Esportare dati articolati per mezzo di regole

```
gr =
 Import["http://mat.gsia.cmu.edu/COLOR/instances/homer.col",
  "DIMACS"]
```

Esportiamo un file TAR compresso composto da due file



Integrazione/interazione con altri ambienti: Import/Export di dati

Una volta importati (o generati) i dati in *Mathematica* bisogna ancora prestare attenzione al modo in cui si intendono gestire ed eventualmente ulteriormente salvare. Facciamo un esempio di ottimizzazione dei tempi di lettura/scrittura di grandi data set.

```
dati = RandomInteger[{-10, 10}, {10^5}];
```

Uno dei formati molto diffusi è il “Comma Separated Value” (CSV). Questo formato è comodo perchè i file sono ASCII e dunque leggibili con qualsiasi editor come Notepad o simili. Lo svantaggio è che sono “pesanti” e rallentano i processi di import export.

```
AbsoluteTiming[Export["dati.csv", dati]]
```

```
{1.110613, dati.csv}
```

```
FileInformation["dati.csv", "ByteCount"]
```

```
257 105
```

```
AbsoluteTiming[t = Flatten[Import["dati.csv"]];]
```

```
{0.995028, Null}
```

Proviamo con il formato Binary

```
AbsoluteTiming[
 stream = OpenRead["dati", BinaryFormat → True];
 tt = BinaryReadList["dati", "Integer16"];
 Close[stream]]
```

```
{0.003106, dati}
```

```
t === tt === dati
```

```
True
```

Un ulteriore vantaggio si ha se i dati in questione vengono frequentemente aggiornati con l’aggiunta di qualche riga.

```
new = RandomInteger[{-10, 10}, {10}];
```

```
AbsoluteTiming[
 stream = OpenAppend["dati", BinaryFormat → True];
 BinaryWrite[stream, new, "Integer16"];
 Close[stream]]
```

```
{0.000141, dati}
```

```
t = Flatten[Import["dati.csv"]];
```

```
AbsoluteTiming[
 stream = OpenRead["dati", BinaryFormat → True];
 tt = BinaryReadList["dati", "Integer16"];
 Close[stream]]
```

```
{0.002640, dati}
```

Integrazione/interazione con altri ambienti: Import/Export di dati

Un altro importantissimo aspetto legato alla gestione dei dati è quello dei database. Anche in questo caso *Mathematica* comprende la tecnologia necessaria per l'integrazione immediata ed altamente flessibile: Database Connectivity

```
(* Esempi *)
```

```
<< DatabaseLink`DatabaseExamples`;
DatabaseExamplesBuild[]
```

Seleziono tutti i dati da una tabella:

```
SQLSelect[conn, "ROYSCHED"]
```

```
BS1011     0     5000   0.1
BS1011   5001   50 000  0.12
CP5018     0     2000   0.1
CP5018   2001    4000   0.12
CP5018   4001   50 000  0.16
BS1001     0     1000   0.1
BS1001   1001    5000   0.12
BS1001   5001    7000   0.16
BS1001   7001   50 000  0.18
PS9999     0    50 000  0.1
PY2002     0     1000   0.1
PY2002   1001    5000   0.12
PY2002   5001   50 000  0.14
PY2003     0     2000   0.1
PY2003   2001    5000   0.12
PY2003   5001   50 000  0.14
UK3004     0     1000   0.1
UK3004   1001    2000   0.12
```

```
Needs["DatabaseLink`"];
conn = OpenSQLConnection["publisher"];
```

```
UK3004   1001    2000   0.12
UK3004   2001    6000   0.14
UK3004   6001    8000   0.18
UK3004   8001   50 000  0.2
CK4005     0     2000   0.1
CK4005   2001    6000   0.12
CK4005   6001    8000   0.16
CK4005   8001   50 000  0.16
CP5010     0     5000   0.1
CP5010   5001   50 000  0.12
PY2012     0     5000   0.1
PY2012   5001   50 000  0.12
PY2013     0     5000   0.1
PY2013   5001   50 000  0.12
UK3006     0     1000   0.1
UK3006   1001    2000   0.12
UK3006   2001    6000   0.14
UK3006   6001    8000   0.18
UK3006   8001   50 000  0.2
BS1014     0     4000   0.1
BS1014   4001    8000   0.12
BS1014   8001   50 000  0.14
UK3015     0     2000   0.1
UK3015   2001    4000   0.12
UK3015   4001    8000   0.14
```

```
UK3015  4001   8000  0.14
UK3015  8001  12000  0.16
CK4016     0   5000  0.1
CK4016  5001  15000  0.12
CK4017     0   2000  0.1
CK4017  2001   8000  0.12
CK4017  8001  16000  0.14
BS1007     0   5000  0.1
BS1007  5001  50000  0.12
PY2008     0  50000  0.1
```

Seleziono solo determinate colonne:

```
Grid[SQLSelect[conn, "ROYSCHED", {"TITLE_ID", "ROYALTY"}]]
```

```
BS1011   0.1
BS1011  0.12
CP5018   0.1
CP5018  0.12
CP5018  0.16
BS1001   0.1
BS1001  0.12
BS1001  0.16
BS1001  0.18
PS9999   0.1
```

```
PY2002   0.1
PY2002   0.12
PY2002   0.14
PY2003   0.1
PY2003   0.12
PY2003   0.14
UK3004   0.1
UK3004   0.12
UK3004   0.14
UK3004   0.18
UK3004   0.2
CK4005   0.1
CK4005   0.12
CK4005   0.16
CK4005   0.16
CP5010   0.1
CP5010   0.12
PY2012   0.1
PY2012   0.12
PY2013   0.1
PY2013   0.12
UK3006   0.1
UK3006   0.12
UK3006   0.14
```

```
UK3006  0.18
UK3006   0.2
BS1014   0.1
BS1014  0.12
BS1014  0.14
UK3015   0.1
UK3015  0.12
UK3015  0.14
UK3015  0.16
CK4016   0.1
CK4016  0.12
CK4017   0.1
CK4017  0.12
CK4017  0.14
BS1007   0.1
BS1007  0.12
PY2008   0.1
```

Unisco i dati da più tabelle:

```
SQLSelect[conn, {"TITLES", "ROYSCHED"},
  {{"TITLES", "TITLE"}, {"TITLES", "TITLE_ID"},
   {"ROYSCHED", "ROYALTY"}},
  SQLColumn[{"TITLES", "TITLE_ID"}] ==
   SQLColumn[{"ROYSCHED", "TITLE_ID"}]] // Short[#, 25] &
```

```
(          Designer Class Action Suits              BS1001   0.1  )
|          Designer Class Action Suits              BS1001   0.12 |
|          Designer Class Action Suits              BS1001   0.16 |
|          Designer Class Action Suits              BS1001   0.18 |
|       Self Hypnosis: A Beginner's Guide           PY2002   0.1  |
|       Self Hypnosis: A Beginner's Guide           PY2002   0.12 |
|       Self Hypnosis: A Beginner's Guide           PY2002   0.14 |
|              Phobic Psychology                    PY2003   0.1  |
|              Phobic Psychology                    PY2003   0.12 |
|              Phobic Psychology                    PY2003   0.14 |
|              Hamburger Again!                     UK3004   0.1  |
|              Hamburger Again!                     UK3004   0.12 |
|              Hamburger Again!                     UK3004   0.14 |
|              Hamburger Again!                     UK3004   0.18 |
|              Hamburger Again!                     UK3004   0.2  |
|    Made to Wonder: Cooking the Macabre            CK4005   0.1  |
|    Made to Wonder: Cooking the Macabre            CK4005   0.12 |
|    Made to Wonder: Cooking the Macabre            CK4005   0.16 |
```

Made to Wonder: Cooking the Macabre	CK4005	0.16
How to Burn a Compact Disk	UK3006	0.1
How to Burn a Compact Disk	UK3006	0.12
How to Burn a Compact Disk	UK3006	0.14
How to Burn a Compact Disk	UK3006	0.18
How to Burn a Compact Disk	UK3006	0.2
Modems for Morons	BS1007	0.1
Modems for Morons	BS1007	0.12
How Green Is My Valley?	PY2008	0.1
Taiwan Trails	CP5010	0.1
Taiwan Trails	CP5010	0.12
Guide to Impractical Databases	BS1011	0.1
Guide to Impractical Databases	BS1011	0.12
Know Thyself	PY2012	0.1
Know Thyself	PY2012	0.12
Where Minds Meat: The Impact of Diet on Behavior	PY2013	0.1
Where Minds Meat: The Impact of Diet on Behavior	PY2013	0.12
Exit Interviews	BS1014	0.1
Exit Interviews	BS1014	0.12
Exit Interviews	BS1014	0.14
Treasures of the Sierra Madre	UK3015	0.1
Treasures of the Sierra Madre	UK3015	0.12
Treasures of the Sierra Madre	UK3015	0.14
Treasures of the Sierra Madre	UK3015	0.16
Too Many Cooks	CK4016	0.1

```
       Too Many Cooks               CK4016   0.1
       Too Many Cooks               CK4016   0.12
     Let Them Eat Cake!             CK4017   0.1
     Let Them Eat Cake!             CK4017   0.12
     Let Them Eat Cake!             CK4017   0.14
Sticky Software: UI and GUI         CP5018   0.1
Sticky Software: UI and GUI         CP5018   0.12
Sticky Software: UI and GUI         CP5018   0.16
```

Seleziono i dati che soddisfano una condizione:

```
SQLSelect[conn, "ROYSCHED", {"TITLE_ID", "ROYALTY"},
 .10 < SQLColumn["ROYALTY"] < .15]
```

$$\begin{pmatrix}
\text{BS1011} & 0.12 \\
\text{CP5018} & 0.12 \\
\text{BS1001} & 0.12 \\
\text{PY2002} & 0.12 \\
\text{PY2002} & 0.14 \\
\text{PY2003} & 0.12 \\
\text{PY2003} & 0.14 \\
\text{UK3004} & 0.12 \\
\text{UK3004} & 0.14 \\
\text{CK4005} & 0.12 \\
\text{CP5010} & 0.12 \\
\text{PY2012} & 0.12 \\
\text{PY2013} & 0.12 \\
\text{UK3006} & 0.12 \\
\text{UK3006} & 0.14 \\
\text{BS1014} & 0.12 \\
\text{BS1014} & 0.14 \\
\text{UK3015} & 0.12 \\
\text{UK3015} & 0.14 \\
\text{CK4016} & 0.12 \\
\text{CK4017} & 0.12 \\
\text{CK4017} & 0.14 \\
\text{BS1007} & 0.12
\end{pmatrix}$$

Seleziono i dati che soddisfano un pattern:

```
SQLSelect[conn, "ROYSCHED", {"TITLE_ID", "ROYALTY"},
 SQLStringMatchQ[SQLColumn["TITLE_ID"], "C%"]]
```

```
CP5018  0.1
CP5018  0.12
CP5018  0.16
CK4005  0.1
CK4005  0.12
CK4005  0.16
CK4005  0.16
CP5010  0.1
CP5010  0.12
CK4016  0.1
CK4016  0.12
CK4017  0.1
CK4017  0.12
CK4017  0.14
```

```
CloseSQLConnection[conn]
```

{SQLConnection(publisher, 1, Open, TransactionIsolationLevel → ReadCommitted)}

◀ | ▶

Integrazione/interazione con altri ambienti: Import/Export di dati

Le banche dati

» GenomeData

Il cromosoma 22

```
GenomeData["Chromosome22Genes"]
```

```
{A26C3, A4GALT, ABCD1P4, ACF, ACO2, ACR, ADM2, ADORA2A, ADPRTL4,
  ADRBK2, ADSL, ADTB1L1, ADTB1L2, AIFM3, ALG12, ANKRD54, ANTP3,
  AP1B1, APOBEC3A, APOBEC3B, APOBEC3C, APOBEC3D, APOBEC3F,
  APOBEC3G, APOBEC3H, APOL1, APOL2, APOL3, APOL4, APOL5, APOL6,
  ARFGAP3, ARHGAP8, ARSA, ARVCF, ASCC2, ASH2LP1, ASLL, ASPHD2,
  ATF4, ATP5L2, ATP6V1E1, ATXN10, BAIAP2L2, BCL2L13, BCR, BCRL2,
  BCRL3, BCRL4, BCRL5, BCRL6, BID, BIK, BMP6P1, BP38, BP55, BPIL2,
BRD1, C1QTNF6, C22CTA250D109, C22orf10, C22orf13, C22orf15, C22orf17,
C22orf23, C22orf24, C22orf25, C22orf26, C22orf27, C22orf28, C22orf29,
C22orf30, C22orf31, C22orf32, C22orf33, C22orf34, C22orf36, C22orf37,
C22orf39, C22orf40, C22orf9, C22RP185F182, CABIN1, CABP7, CACNA1I,
CACNG2, CARD10, CBX6, CBX7, CBY1, CCDC116, CCDC117, CCDC134,
CCT8L2, CDAGS, CDC42EP1, CDC45L, CECR, CECR1, CECR2, CECR3,
CECR4, CECR5, CECR6, CECR7, CECR8, CECR9, CELSR1, CENPM, CERK,
CHADL, CHCHD10, CHEK2, CHKB, CHKBCPT1B, CLCP1, CLDN5, CLTCL1,
CN5H64, COMT, COPD4, COX5BL7, COX6BP3, COX7BP1, CPSF1P1,
CPT1B, CRELD2, CRKL, CRYBA4, CRYBB1, CRYBB2, CRYBB2P1,
CRYBB3, CSDC2, CSF2RB, CSF2RB2, CSNK1E, CTA126B43, CTA216E106,
CTA221G94, CTA229A82, CYB5R3, CYP2D6, CYP2D7P1, CYP2D7P2,
CYP2D8P1, CYP2D8P2, DDT, DDTL, DDX17, DEPDC5, DERL3, DFNB40,
DGCR, DGCR10, DGCR11, DGCR12, DGCR14, DGCR2, DGCR5, DGCR6,
DGCR6L, DGCR7, DGCR8, DGCR9, DJ1033E151, DJ222E132, DKFZp434K191,
DKFZP434P211, DKFZp547B139, DMC1, DNAJB7, DNAL4, DNM1DN18,
DRG1, DUSP18, DUXAP8, DVL1L1, EFCAB6, EIF3D, EIF3EIP, EIF4ENIF1,
ELFN2, EMID1, ENTHD1, EP300, EWSR1, FABP5L11, FAM108A5, FAM109B,
FAM116B, FAM118A, FAM152B, FAM19A5, FAM32B, FAM83F, FBLN1,
FBXO7, FLJ20464, FLJ23185, FLJ23865, FLJ26056, FLJ27365, FLJ30901,
FLJ32575, FLJ32756, FLJ34651, FLJ38343, FLJ39582, FLJ42953, FLJ43315,
FLJ44385, FLJ90680, FOXRED2, FPEVF, FRA22A, FRA22B, GAB4, GAL3ST1,
GALR3, GAS2L1, GCAT, GGA1, GGT1, GGT2, GGT3P, GGT5, GGTLC2,
GGTLC3, GGTLC4P, GGTLC5P, GNAZ, GNB1L, GP1BB, GRAMD4, GRAP2,
GSC2, GSTT1, GSTT2, GSTT2B, GSTTP1, GSTTP2, GTPBP1, GTSE1, H1F0,
HCG17241, HDAC10, HIC2, HIRA, HMG17L1, HMG17L2, HMG1L10,
HMG2L1, HMOX1, HORMAD2, HPS4, HSCB, HTF9C, HUMYZ82H07,
IGKV1OR221, IGKV1OR225, IGKV2OR223, IGKV2OR224, IGKV3OR222,
IGL@, IGLC@, IGLC1, IGLC2, IGLC3, IGLC4, IGLC5, IGLC6, IGLC7,
IGLCOR221, IGLCOR222, IGLJ@, IGLJ1, IGLJ2, IGLJ3, IGLJ4, IGLJ5, IGLJ6,
IGLJ7, IGLL1, IGLL2, IGLL3, IGLV@, IGLV1054, IGLV1067, IGLV1155,
IGLV136, IGLV140, IGLV141, IGLV144, IGLV147, IGLV150, IGLV151,
IGLV162, IGLV211, IGLV214, IGLV218, IGLV223, IGLV228, IGLV233,
IGLV234, IGLV25, IGLV28, IGLV31, IGLV310, IGLV312, IGLV313, IGLV315,
IGLV316, IGLV317, IGLV319, IGLV32, IGLV321, IGLV322, IGLV324,
IGLV325, IGLV326, IGLV327, IGLV329, IGLV330, IGLV331, IGLV332,
IGLV34, IGLV36, IGLV37, IGLV39, IGLV43, IGLV460, IGLV469, IGLV537,
IGLV539, IGLV545, IGLV548, IGLV552, IGLV657, IGLV735, IGLV743,
IGLV746, IGLV861, IGLV949, IGLVI20, IGLVI38, IGLVI42, IGLVI56,
IGLVI63, IGLVI68, IGLVI70, IGLVIV53, IGLVIV59, IGLVIV64, IGLVIV65,
IGLVIV661, IGLVIVOR221, IGLVIVOR222, IGLVV58, IGLVV66, IGLVVI221,
IGLVVI251, IGLVVII411, IL17RA, IL17REL, IL2RB, ISX, ITS, JOSD1,
KAZA, KCNJ4, KCNMB3L, KCTD17, KDELR3, KIAA1648, KIAA1652,
KIAA1654, KIAA1655, KIAA1656, KIAA1659, KIAA1661, KLHDC7B,
KLHL22, KREMEN1, KRT18P23, KRT18P5, L3MBTL2, LARGE, LDOC1L,
LGALS1, LGALS2, LIF, LIMK2, LL22NC0375B36, LMF2, LOC100113375,
LOC100127979, LOC100128009, LOC100128138, LOC100128147, LOC100128279,
LOC100128388, LOC100128393, LOC100128394, LOC100128400, LOC100128401,
LOC100128525, LOC100128526, LOC100128530, LOC100128531, LOC100128533,
LOC100128535, LOC100128677, LOC100128716, LOC100128943, LOC100128946,
LOC100129113, LOC100129224, LOC100129225, LOC100129254, LOC100129262,
LOC100129358, LOC100129359, LOC100129360, LOC100129389, LOC100129643,
LOC100129647, LOC100129648, LOC100129803, LOC100129936, LOC100129939,
LOC100129944, LOC100130079, LOC100130194, LOC100130269, LOC100130338,
LOC100130339, LOC100130420, LOC100130493, LOC100130496, LOC100130558,
LOC100130561, LOC100130624, LOC100130708, LOC100130709, LOC100130715,
LOC100130773, LOC100130828, LOC100130899, LOC100131052, LOC100131298,
LOC100131299, LOC100131314, LOC100131322, LOC100131346, LOC100131404,
LOC100131405, LOC100131424, LOC100131530, LOC100131536, LOC100131537,
LOC100131579, LOC100131620, LOC100131667, LOC100131702, LOC100131753,
LOC100131802, LOC100131861, LOC100131872, LOC100132029, LOC100132032,
LOC100132082, LOC100132173, LOC100132176, LOC100132273, LOC100132295,
LOC100132389, LOC100132451, LOC100132621, LOC100132705, LOC100132706,
LOC100132723, LOC100132900, LOC100133040, LOC100133041, LOC100133043,
LOC100133044, LOC100133064, LOC100133143, LOC100133163, LOC100133167,
LOC100133224, LOC100133475, LOC100133478, LOC100133614, LOC100133843,
LOC100133850, LOC100133887, LOC100133919, LOC100133922, LOC100134027,
LOC100134031, LOC100134074, LOC100134107, LOC100134194, LOC100134361,
LOC100134400, LOC100134402, LOC100134421, LOC129026, LOC150166,
LOC150185, LOC150287, LOC150381, LOC150417, LOC246785, LOC284861,
LOC284865, LOC284889, LOC284890, LOC284898, LOC284912, LOC284926,
LOC284930, LOC284933, LOC284939, LOC339666, LOC339674, LOC339685,
LOC376818, LOC386610, LOC388849, LOC388882, LOC388885, LOC388889,
LOC388906, LOC388907, LOC388915, LOC391322, LOC391334, LOC400879,
LOC400891, LOC400927, LOC402036, LOC402057, LOC440786, LOC440792,
LOC440795, LOC440817, LOC440821, LOC440836, LOC441996, LOC51233,
LOC550631, LOC553158, LOC554174, LOC641379, LOC641457, LOC642619,
LOC642633, LOC642643, LOC642648, LOC643069, LOC643102, LOC643653,
LOC644079, LOC644165, LOC644186, LOC644339, LOC644525, LOC644618,
LOC644652, LOC644768, LOC644773, LOC644899, LOC645280, LOC645348,
LOC645390, LOC645904, LOC645964, LOC646074, LOC646216, LOC646408,
LOC646467, LOC646756, LOC646851, LOC648218, LOC648691, LOC648771,
LOC648980, LOC649017, LOC649095, LOC649264, LOC651158, LOC652968,
LOC653135, LOC653184, LOC653203, LOC653264, LOC653881, LOC692231,
LOC692246, LOC727731, LOC727962, LOC727983, LOC727996, LOC728139,
LOC728212, LOC728216, LOC728253, LOC728351, LOC728418, LOC728438,
LOC728451, LOC728468, LOC728474, LOC728877, LOC728897, LOC729057,
LOC729166, LOC729212, LOC729314, LOC729444, LOC729461, LOC729526,
LOC729568, LOC729832, LOC729980, LOC730005, LOC730163, LOC730668,
LOC730990, LOC731062, LOC731957, LOC768328, LOC780780, LOC90834,
LOC91219, LOC91316, LOC96610, LRE1, LRP5L, LZTR1, MAFF, MAP3K7IP1,
MAPK1, MAPK11, MAPK12, MAPK8IP2, MB, MCAT, MCHR1, MCM5,
MED15, MEI1, MFNG, MGAT3, MGC15705, MGC16703, MGC20647,
MGC70863, MGCR, MIAT, MICAL3, MICALL1, MIF, MIOX, MIRN130B,
MIRN185, MIRN301B, MIRN33A, MIRN648, MIRN649, MIRN650, MIRN658,
MIRN659, MIRNLET7A3, MIRNLET7B, MKL1, MLC1, MMP11, MN1,
MORC2, MOV10L1, MPPED1, MPST, MRPL40, MRPS16P3, MRPS18CP6,
MTMR3, MTP18, MYH9, MYO18B, MYP6, NAGA, NCAPH2, NCF4,
NDUFA6, NDUFA9P1, NEFH, NF1L6, NF2, NFAM1, NHP2L1, NIPSNAP1,
NOL12, NPTXR, NUP50, OGS2, OPA5, OPHN1P1, OR11H1, OSBP2, OSM,
P2RX6, P2RX6P, PACSIN2, PANX2, PARVB, PARVG, PATZ1, PDGFB,
PDXP, PES1, PEX26, PHF21B, PHF5A, PI4KA, PI4KAP1, PI4KAP2, PIB5PA,
PICK1, PIK3IP1, PIM3, PISD, PITPNB, PIWIL3, PKDREJ, PLA2G3, PLA2G6,
PLXNB2, PMM1, PNPLA3, PNPLA5, POLDIP3, POLR2F, POLR3H, POM121L1,
POM121L4P, PPARA, PPIL2, PPM1F, PPP1R14BP1, PRAME, PRAMEL,
PRODH, PRR5, PSCD4, PsiTPTE22, PTPNS1L, PVALB, RAB36, RABL2B,
RABL4, RAC2, RANBP1, RANGAP1, RASD2, RASL10A, RBM9, RBX1,
RFPL1, RFPL1S, RFPL2, RFPL3, RFPL3S, RGL4, RHBDD3, RIBC2, RIMBP3,
RIMBP3B, RIMBP3C, RN7SLP3, RNF185, RNF215, RNR5, RNU12P, RNU13P1,
RNU86, RP11191L91, RP1127L46, RP1199H161, RP3402G115, RP3474I125,
RPL3, RPL32L2, RPL41P3, RPS17P4, RPS19BP1, RPS9P2, RTDR1, RTN4R,
SAMM50, SAPS2, SBF1, SCARF2, SCO2, SCUBE1, SCZD4, SDC4P, SDF2L1,
SEC14L2, SEC14L3, SEC14L4, SELM, SEPT3, SEPT5, SERHL, SERHL2,
SERPIND1, SEZ6L, SF3A1, SFI1, SGSM1, SGSM3, SH3BP1, SHANK3,
SHFM3P1, SLC16A8, SLC25A1, SLC25A17, SLC25A18, SLC25A5P1,
SLC2A11, SLC35E4, SLC5A1, SLC5A4, SLC7A4, SLC9A3P2, SMARCB1,
SMC1B, SMCR7L, SMTN, SNAP29, SNORD125, SNORD43, SNORD83A,
SNORD83B, SNRPD3, SOCS2P2, SOX10, SPECC1L, SREBF2, SRRD, SSTR3,
ST13, SULT4A1, SUSD2, SUSD2P1, SYN3, SYNGR1, TBC1D10A, TBC1D22A,
TBX1, TCF20, TCN2, TEF, TERF2IPP1, TFIP11, THAP7, THOC5, TI227H,
TIMP3, TMEM184B, TMEM191A, TMEM191B, TMEM191C, TMEM211,
TMPRSS6, TNFRSF13C, TNRC6B, TOB2, TOM1, TOMM22, TOP1P2, TOP3B,
TOP3B2, TPST2, TRABD, TRIOBP, TRMU, TRNASTOPUCA, TRSPP1,
TSPO, TSSK1A, TSSK2, TST, TTC28, TTC38, TTLL1, TTLL12, TTLL8,
TUBA8, TUBGCP6, TUG1, TXN2, TXNRD2, TYMP, UBE2L3, UCRC,
UFD1L, UNC84B, UNQ1945, UNQ6126, UPB1, UPK3A, UQCRFSL1, USP18,
USP41, VCF, VN1R9P, VPREB1, VPREB3, VWFP, WBP2NL, WNT7B,
XBP1, XKR3, XPNPEP3, XRCC6, YDJC, YESP, YME1L2, YPEL1, YWHAH,
YWHAQP2, ZBED4, ZC3H7B, ZD77D08, ZDHHC8, ZDHHC8P, ZMAT5,
ZNF280A, ZNF280B, ZNF402P, ZNF70, ZNF72, ZNF73, ZNF74, ZNRF3}
```

Un particolare gene di questo cromosoma

```
GenomeData["C22CTA250D109"]
```

```
ATGGCGCGGGCCGGGGCGCGGGGACTGCTCGGCGGCCGCCGTCCTCCCGGC⋱
    CTCCGGCTCGCGCTCGCGCTTCGGCTCGCGTTGCTGCTGGCGCGGCCGC⋱
    CGTCGGGCCGCGCGGGAGCCCCCGAGGCGCAGGGTCCCGCGGCGCCC⋱
    GGCACGACAGCCCCGGAGGGGGGCGACCGCTGCCGCGGCTACTACGA⋱
```

```
CGTGATGGGCCAGTGGGACCCGCCCTTCAACTGCAGCTCCGGAGCCTA⋱
CAGCTTCTGCTGCGGCACGTGCGGCTACCGCTTCTGCTGCCACGACGG⋱
GCCGCGGCGCCTCGACCAGAGCCGCTGTTCCAACTACGACACGCCGGC⋱
CTGGGTCCAGACAGGCCGGCCGCCCGCCCGCGCCCGCGACACCGCAGC⋱
GCCCCGGGACCCCGGCCGCGAGCGCAGCCATACGGCCGTCTACGCTGT⋱
GTGCGGCGTCGCAGCGCTGCTGGTGCTGGCCGGCATCGGGGCGCGCCT⋱
GGGACTGGAGAGGGCGCACAGCCCGCGCGCGGCGCACAGTGACCA⋱
GGTGAGCACGCGCCGCCCAGGGACCCCCGCCTGCTCCTCGGACTGCTC⋱
TCCCTGCCCGCTGATCCTTCCCGCCGCTCTCGAAGCGCTGGTTCTCCCA⋱
GCAGGCCCCTTCTCTGGGTTTTCCTGTCTCCTTCTCCCCCCGCCGCCCCC⋱
CCAACACTTCATCAGGGAGACACCTGACACCGCCCTGACTCCCTTTGA⋱
CCTTCTGCCCGTGCTGGGCGCCGCACCGCCTTTCCCTGCTCGAGTCTCC⋱
TCTGGAGGGCAGGGGGGTGGGGGCGGGCTGGGAGGGGGACAGGTAGC⋱
CTTGCACAGAGGGCACCGGGTCACTGTCCAGGGTAAGGGGCGCCGCTG⋱
GCCAGGGAGGTGACCGGGCGTCTTGGGCACGGGAAAGGAAGGGGCGC⋱
TTGGTTCTCTGGGTGCCCATCTCCTGCTTCTCCCCACTCCTCCCTCCCCT⋱
CTTCCACCTCCTGCCTCTCCCCTGCCCCCACAGCTGTGACTCAAGGCTG⋱
CCCTTCCCTCCTTGGGTTTCAAAGGCCTCCCCCCACTCACCCCATGCCA⋱
CAGCTCCCAGGTGAAGGTCAGGGCTCCCTCTGCGTAGAAAGGGGACTC⋱
TGGAACACAGTCGTGTGTTGCTGAGGCCAGGACAGGTGCTTGTTGCAT⋱
TCTGAGCTTGGTGTGAAGTGGGCACTACAGGGTCCAGGTCCCCTTCCTC⋱
GGTTGGAACGAGGACTCTAGGACTCTATGTGTCCTGACCCTGGAACTC⋱
CCACCCACATGCCACACGTGGCACTCACATGCCCCCAGCAACCCCCAT⋱
```

```
GCCCTGGGGCCTGTGCCCTCCTCATCCACCCGGTCCCAGCACCCTTTGG⋱
GCCTGTGGGCTGCTGACACCGAATGTGACAGGCTGGTATTTAAAAGTG⋱
ACCATAGAGAAGCCTTCATACCCTGGCAGGCCAGATGTGGACACTTCC⋱
CCCTCCTGGGCCTCAGAGGCAGGGGGTGAGGGCTTCCTTGGCACTTCA⋱
GCACCCCTCGCTCCCGAAACCACGTAGCCCACTTTCTCCCAGGGAGCTC⋱
TAAGGGAAGCCCGGGCTGGCACCCTGGCCTCCACCCTCAGCTTCTGCA⋱
CTGCCTTCAGGAGGTGGAGCCTCCTGGCCTGGCTTCTGGATGCTGCAG⋱
GGGTAGGGGCTGGGCTGCCCAAATTTGGGGGTCCCCCATGATGGGTGA⋱
GGGTAGCTTGGAAGAGGCCTGGATATGTCTTTGACAGGGGCGGGGAAA⋱
AAGGAACTTAGGGCTGCGATGGTGAGGGGTGGCTGCAGCAGCCCAGCT⋱
GACATGTTCATCCTGAACTCCCTGCACCTGCTGGGGATGGGGCAGCAG⋱
GTTTGGTGATGCCGCAGGAAAGCCATGCCCACCCTTGCTGTGCTGTGTG⋱
GTGTAGGCCAGTCCCTTCTCCTTTCTGGGCTTCAGGCTCCCATGGCCCC⋱
AACTCCCTTTTCTCTGCTGGGGAGACCTGGAATGGGACGTAGGCTTGTG⋱
TGTGGGTGTGTCCTGAGTCCCTGTCCTCCCCTCGAGTCTCACCCTTCATT⋱
CTGCTGTGTTGGGGTTGAAGGCCAGGGAAGGCTTCTCAGAGGGCTTGG⋱
GCGGGGCATCTGGAAAGAGTTTGTGCAAAGGCCCAGAGGCGTGGTGGT⋱
TACTCAGCTGTCAGGGAGACAGGAGCTTGAGGTGCTTGCCCAGAGATG⋱
AGGCCACAGAGGTGAGCCTCTGGGGAGAGGGCAGGGCAGGAAGGTCA⋱
GCCCAGGCAGGTACTAGGATGGTTTCAGACCCTGCCACCAAACCTGAA⋱
GCCAAACCTCGGCCATCCCTGGTGTCCTGCCTCCTGAATGACCTTCACG⋱
AATCAGCCATTTGAGGCTCTAGCCCCGGAAGGAGGGTCAGGCTGAGTG⋱
GGAGAAAATCATGGGTTCCGGGTTCTGAAGGCAGCTGGGCTGCAGACA⋱
```

```
CTGTCACATCACAGCTGGGTGGGGGCTGGGTGTTGGGTCCAAGGAGGT⋱
GGCTGTCCAAGGTCACACAGCCTGTGGGTCCTGGGCTGCCACCAGGGG⋱
GTCTCCCAACATCCCCATTCCGTCACCCATCTGTCCTTCATGCCCGTGA⋱
TGGCAGCTTTATCCCAGGTCCCAGCAGAAGCCGCGGCTGCAGGGGCAG⋱
TTAGGCAGGCAGGTAAATAAGGCATCCTGGGCCGGGCGCGGTGGCTCA⋱
CACCTGTCATCTCAGCACTTTGGGAGGCCGAGACCGGCGGAGTTGGAG⋱
ACCAGCCTGGCCAACATGGTGAAACCCCGTCTCTACTAAAAATACAAA⋱
AATTAGCTAGGTGTGGTGGCAGGCACCTTATAATCCCAGCTACTAGGG⋱
AAGCTGAGGCTGGAGAATCACTTGAACCGGGGAGGTGGAGGTTGCAG⋱
TGAGCCAAGATTGCGCCATTGCACTCCAGCCTGGACAACAGAGCGAGA⋱
CTCTGCTTCAAAAGATAAAAAAATTAAAAAAAAAGGATTCCTGGAGGAG⋱
GCCGCTGATGGGGTCTGAAGAATGAGTAGCAGTTTTTTGGGAGAAGAG⋱
GCAGGAAAGCAGAGAAAGCTCGCATTGGGAGGGGAGGCTCACCTGGG⋱
GAAGGTCCCTGAGAGTGGGCTCCAGGGAGGGGAGGGGAGGGAAGGGG⋱
AGCCTCATTAGAGTTTCCTGACCTGCTCTGGGAAGCAGCTGTCTTGGGG⋱
CCACAGGTGAGGCTGGCAGATACTGCCTGGGAGGGGAAACTCGTGCGT⋱
GATTTCGAGCACCGGAGAGGACGGCCTGGAGAGGCCGAAAGAACTGC⋱
CCAGGGCCACACGGTGAGACGGTGGTGGCCGGGGCTGGCTTTTGATGA⋱
GGTGGGAGACAGAGGGCCACCCCTGACTGCTGTCCCCGCAGGGCGCTG⋱
ACAGAGCTTCTGAAGCAGCCGGGCCCCCAGGAGCCACTGCCTCCCACC⋱
CTGGGCCCACCCCTGGGTGGCTGTGTCCAGGTGCAGATGGGGGACGGC⋱
CTCCCCCGGGGCTCCCCCCACAACAGCGCAGGTGAGTCGGTGGCGGGC⋱
GGGGCTGAGCGGCGGCGGAGCACGCGGAGGGGCGTGAGAGGCCGAGA⋱
```

```
GGCGGTCCCTCGGCTGGGGCATAGAGGCTTCCCCGGTGGAAATGAGCC⋱
GCGGCCTCTGCGGGTGAGGGGCCCGACCGGCCAGCCAGCTCGCCAGGG⋱
TGGAAAGGGACCCCCCACGGGTCCTGCGGCCTCGAGTCTGGGGGGCTG⋱
TGATGTGTGAGGAAGGTTCCAGATGCAGGATTTGCTGGCCGGGGCTGT⋱
GATCTGAGGCTCGGGCTCCAGAGCCTCCCCCTGGGTCTCTCGAGAGGC⋱
AGGCAGGCTTGGCAGCCCCCACGCCCCGTCTGGGTTCTGTTCCTCAGAC⋱
ATCCCCCGTCTCGGCGGGAGCAGGTCCCAGGACTCCGCGCCTGGCTCG⋱
AGCGTCGCCTCCGGGCGAGCCGTTCATGAGGGTGGCACCCCCCGGGCT⋱
GGCTGCCGCCGCCGCTGCGCGTGACTCGGAGCCAGGCCCTGTTCCAGG⋱
CGCTGGAGGCTGCGGCGAAGACAGGCTTGGCTCGCAGCCTCGCCCGGA⋱
GAACGGTCCCCCAGGTTCTCACCGAACGGCCTCGGGGAGAGGACTACG⋱
GAGAGGTGACAGCGAGGGCGGAGTGAACCGGGGCATCGGACCGAGCC⋱
GCGTCTCACTCCTCGCTCCAGACAAGAAGCGCCTCAACAACGCGCCCC⋱
GGGGGTCGGCCGCCCCGGGGCCCCCGCGCGGCCCGCGGCTGCAGGGC⋱
GGCGGCAGCCTGACGCTGCAGCCAGACTACGCCAAGTACGCCACGTTC⋱
AAGGCCGCCGCGCTCAAGGCCGCAGGTGAGTGGCGGGTGCGGGCAGC⋱
GTCAGGGCACCTGGGCGCACGAGCTCCCGGACTGCGAAGCGCGGGGC⋱
CGCAGCTCCAAGCGCCAAATTCCCAGTTCCGCGCGCCCGCCGCTCGGC⋱
CAATCGGCTCCCTCGCCCCAGCGGGCCCCGCCCCCGGCCCGCCCCGGC⋱
CCGGTCCCCGTCCCAGTCCCCGGCCGGGACCCCTTGTCCAGTCCTGAGC⋱
TTGGGTGCTCGGCGCGGCGTCACTCTTCCCTGCCCTGTCCCCGCAGAGG⋱
CCGCCCCGCGGGACTTCTGTCAGCGTTTCCCCGCCCTCGAGCCGTCCCC⋱
GCGGCAACCCCCGGCGCGGGCTCCGCGACCATCCCCGGACTTGCCTGC⋱
```

```
GCCGCTGGACGCCTGCCCCTGGGCCCCGCCGGTCTACGCGCCCCCTGC⋱
CGCGCCGGGCCCCTATGCCGCCTGGACCTCCAGTCGCCCGGCCCGGCC⋱
CGCCCCGCTCAGCCACCCGACGGCTCGGGCCTTCCAGGTACCCCGGCG⋱
ACCCGGGCACGCGGCCCGGCGCCAGTTCAGTGTGAAGATGCCTGAGAC⋱
CTTCAACCCGCAGCTCCCCGGCCTTTACGGCAGCGCGGGCCGCGGGTC⋱
CCGGTACCTAAGGACCAATAGCAAGACCGAGGTCACCGTGTGA
```

```
GenomeData["C22CTA250D109", "SequenceLength"]
```

```
4903
```

Trovare i 15 geni più corti nel genoma umano

```
Take[Sort[{GenomeData[#, "SequenceLength"], #} & /@
   GenomeData[]], 15]
```

$$\begin{pmatrix} 11 & \text{IGHD727} \\ 16 & \text{IGHD411} \\ 16 & \text{IGHD417} \\ 16 & \text{IGHD44} \\ 17 & \text{IGHD11} \\ 17 & \text{IGHD114} \\ 17 & \text{IGHD120} \\ 17 & \text{IGHD17} \\ 18 & \text{IGHD66} \\ 19 & \text{IGHD423} \\ 20 & \text{IGHD126} \\ 20 & \text{IGHD518} \\ 20 & \text{IGHD524} \\ 20 & \text{IGHD55} \\ 21 & \text{IGHD613} \end{pmatrix}$$

`GenomeLookup` permette di trovare la posizione di sequenze di DNA nel genoma umano:

```
Chromosome1     108 939 073
1               108 939 087
Chromosome1     138 309 610
-1              138 309 624
Chromosome5     139 640 264
-1              139 640 278
Chromosome8     72 019 948
1               72 019 962
Chromosome9     110 092 060
1               110 092 074
```

```
GenomeLookup["CTCTCTAACTAAACT", "Count"]
```

```
5
```

» AstronomicalData

La densità dei pianeti

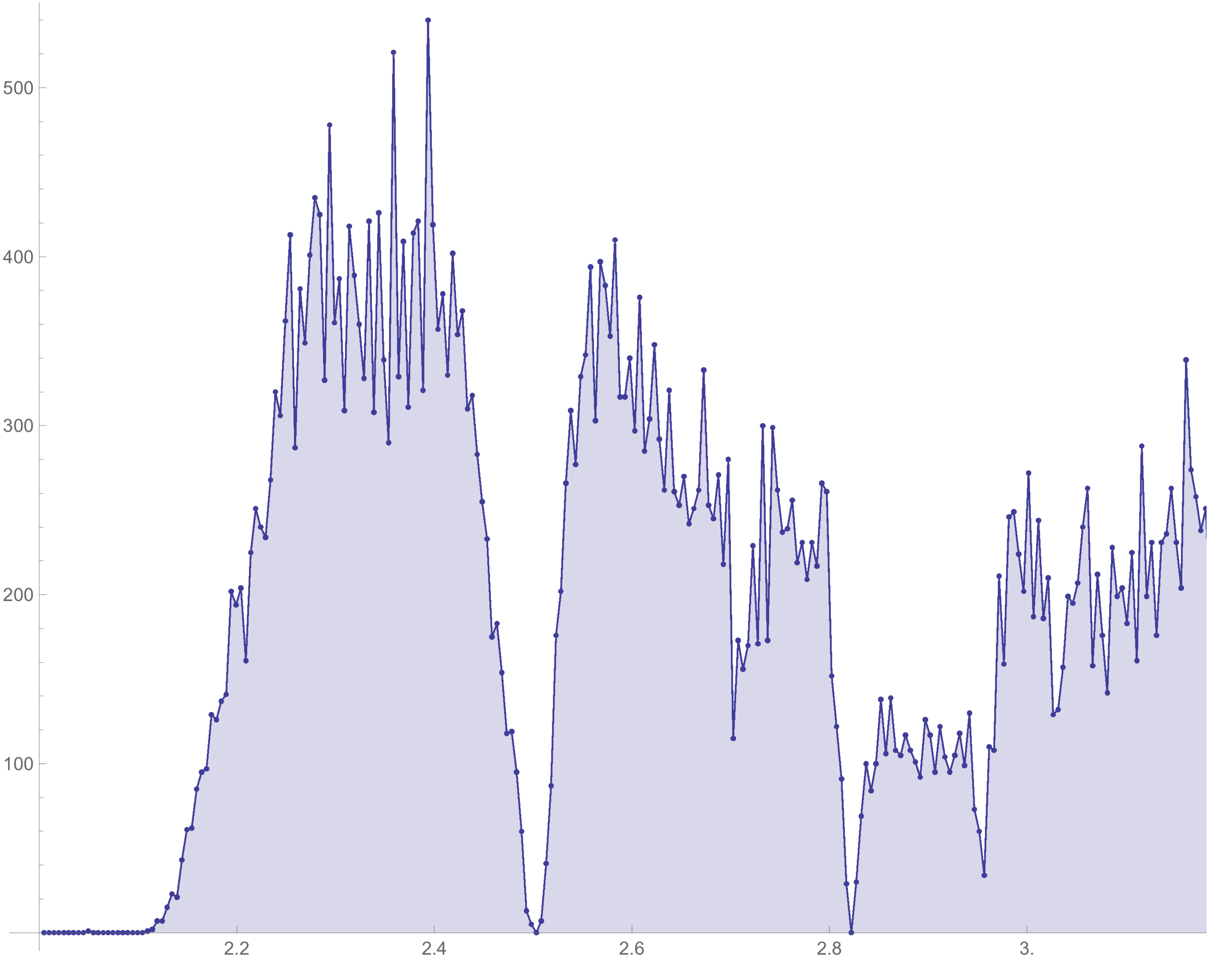
Mercury	5427.
Venus	5243.
Earth	5515.
Mars	3934.0
Jupiter	1326.2
Saturn	687.1
Uranus	1270.
Neptune	1638.

Computa e visualizza la distribuzione dei pianeti minori a varie distanze dal sole:

```
Text[
 Grid[
  {AstronomicalData[#, "Name"],
     AstronomicalData[#, "Density"]} & /@
   AstronomicalData["Planet"], Frame → All,
  Background → LightYellow]]
```

```
ListPlot[asteroidCount, Joined → True, Filling → 0,
 Mesh → All,
 Ticks → {Table[{Rescale[x, {2, 3.5}, {0, 301}], x},
     {x, 2, 3.5, .2}], Automatic}]
```

```
asteroidCount =
  BinCounts[
   Sort@
    Cases[
     (AstronomicalData[#, "SemimajorAxis"] /
          149 597 870 691) & /@
      Join[AstronomicalData["InnerMainBeltAsteroid"],
       AstronomicalData["MainBeltAsteroid"],
       AstronomicalData["OuterMainBeltAsteroid"]],
     x_?NumberQ], {2, 3.5, .005}];
```


500
400
300
200
100
2.2
2.4
2.6
2.8
3.

CountryData

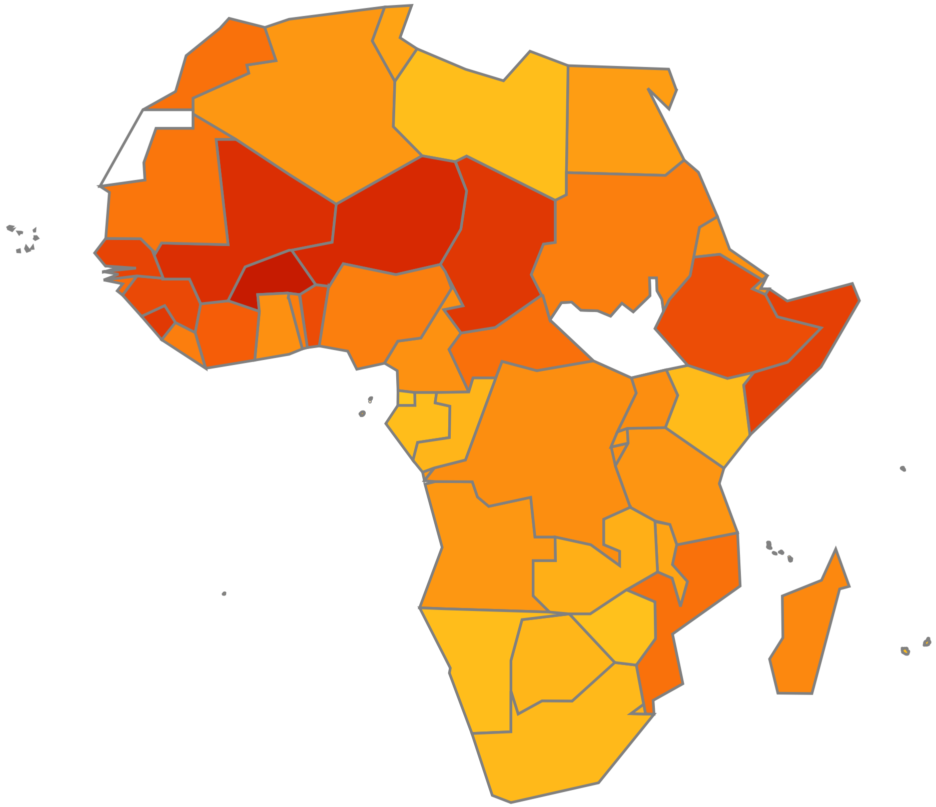
I 20 paesi più grandi per estensione del territorio

```
Last /@
 Take[
  Reverse[
   Sort[{CountryData[#, "Area"], #} & /@ CountryData[]]], 20]
```

```
{Russia, Canada, UnitedStates, China, Brazil, Australia, India,
  Argentina, Kazakhstan, Algeria, DemocraticRepublicCongo, Greenland,
  Mexico, SaudiArabia, Indonesia, Sudan, Libya, Iran, Mongolia, Peru}
```

	area	population
Russia	1.70752×10^{7}	1.42558×10^{8}
Canada	9.98467×10^{6}	3.49937×10^{7}
United States	9.63142×10^{6}	3.1933×10^{8}
China	9.59696×10^{6}	1.35937×10^{9}
Brazil	8.51488×10^{6}	2.0005×10^{8}

Tasso di alfabetizzazione in alcuni continenti (Africa, Europa e Asia):



```
Graphics[
 {EdgeForm[Gray],
    Catch[ColorData["Warm"][
      CountryData[#, "LiteracyFraction"] /.
       _Missing ⧴ Throw[White]]],
    CountryData[#, "SchematicPolygon"]} & /@
  CountryData["Europe"]]
```

```
Graphics[
 {EdgeForm[Gray],
    Catch[ColorData["Warm"][
      CountryData[#, "LiteracyFraction"] /.
       _Missing ⧴ Throw[White]]],
    CountryData[#, "SchematicPolygon"]} & /@
  CountryData["Asia"]]
```

Il grafico dei paesi confinanti in Europa:

```
GraphPlot[
 Flatten[
  Thread[# -> CountryData[#, "BorderingCountries"]] & /@
   CountryData["Europe"]], VertexLabeling → True]
```

Una mappa con i nomi di tutte le nazioni: