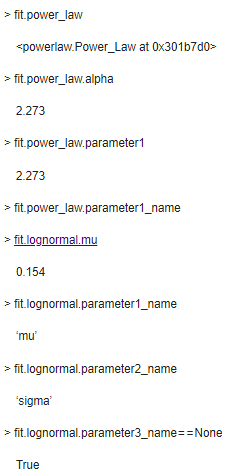
Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif.

Dalam tulisan ini, kami akan memperkenalkan karakteristik Python. Python adalah bahasa pemrograman tingkat tinggi yang bertujuan umum dan banyak digunakan dalam waktu belakangan ini [1] [2] [3]. Filosofi desainnya menekankan kemampuan kode, dan sintaksnya memungkinkan programmer untuk mengekspresikan konsep dalam lebih sedikit baris kode daripada yang mungkin dalam bahasa seperti C [4]. Konstruk bahasa memungkinkan pengguna untuk menulis program yang jelas baik dalam skala kecil maupun besar [5]

**Comparing Candidate Distributions**

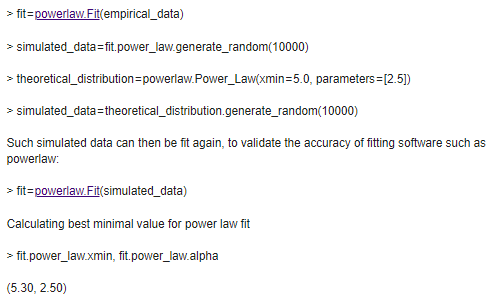
From the created Fit object the user can readily access all the statistical analyses necessary for evaluation of a heavy-tailed distribution. Within the Fit object are individual Distribution objects for different possible distributions. Each Distribution has the best fit parameters for that distribution (calculated when called), accessible both by the parameter's name or the more generic “parameter1”. Using the blackout data:



Goodness of fit dari distribusi ini harus dievaluasi sebelum menyimpulkan bahwa undang-undang kuasa adalah deskripsi data yang baik. Goodness of fit untuk setiap distribusi dapat dipertimbangkan secara individual atau dengan membandingkannya dengan fit dari distribusi lain (masing-masing, menggunakan bootstrap dan uji Kolmogorov-Smirnov untuk menghasilkan nilai-p untuk kecocokan individu vs menggunakan rasio kemungkinan log untuk mengidentifikasi yang mana dari dua cocok lebih baik) [5]. Ada beberapa alasan, baik yang praktis maupun filosofis, untuk fokus pada tes perbandingan terakhir.

**Creating Simulated Data**

Membuat data simulasi yang diambil dari distribusi teoretis sering kali berguna untuk berbagai tugas, seperti pemodelan. Objek Distribusi individual dapat menghasilkan titik data acak dengan fungsi generate\_random. Objek Distribusi ini dapat dipanggil dari objek Fit atau dibuat secara manual

.

**Perhitungan dan Estimasi Distribusi Diskrit**

Sementara kemungkinan maksimum yang cocok dengan hukum kekuasaan berkelanjutan untuk suatu yang diberikan dapat dihitung secara analitis, dan dengan demikian parameter yang dipasang optimal dan dihasilkan dapat dihitung dengan cepat, ini tidak demikian untuk kasus diskrit. Kemungkinan kecocokan maksimum untuk hukum daya diskrit ditemukan oleh optimisasi numerik, yang perhitungannya untuk setiap nilai yang mungkin membutuhkan waktu. Untuk menghindari masalah ini, powerlaw dapat menggunakan estimasi analitik, dari [5], yang dapat "memberikan hasil yang akurat hingga sekitar 1% atau lebih baik disediakan" saat tidak menggunakan. Opsi estim\_discrete ini Benar secara default. Kembali ke data pemadaman:

**Jupyter Notebook**

 Jupyter Notebook adalah aplikasi web sumber terbuka yang dapat Anda gunakan untuk membuat dan berbagi dokumen yang berisi kode langsung, persamaan, visualisasi, dan teks. Notebook Jupyter dikelola oleh orang-orang di Project Jupyter.

Jupyter Notebooks adalah proyek spin-off dari proyek IPython, yang dulu memiliki proyek IPython Notebook itu sendiri. Nama itu, Jupyter, berasal dari bahasa pemrograman inti yang didukung yang didukungnya: Julia, Python, dan R. Jupyter dikirimkan dengan kernel IPython, yang memungkinkan Anda untuk menulis program-program Anda dengan Python, tetapi saat ini terdapat lebih dari 100 kernel lain yang Anda bisa juga digunakan.

**Installation**

Jika demikian, maka Anda dapat menggunakan alat praktis yang dilengkapi dengan Python disebut pip untuk menginstal Jupyter Notebook seperti ini:



Distribusi Python terpopuler berikutnya adalah Anaconda. Anaconda memiliki alat penginstal sendiri bernama conda yang dapat Anda gunakan untuk menginstal paket pihak ketiga. Namun, Anaconda dilengkapi dengan banyak perpustakaan ilmiah yang sudah diinstal, termasuk Jupyter Notebook, jadi Anda tidak perlu melakukan apa pun selain menginstal Anaconda itu sendiri.

### Starting the Jupyter Notebook Server

Sekarang Anda sudah menginstal Jupyter, mari kita pelajari cara menggunakannya. Untuk memulai, yang perlu Anda lakukan adalah membuka aplikasi terminal Anda dan pergi ke folder pilihan Anda. Saya sarankan menggunakan sesuatu seperti folder Documents Anda untuk memulai dan membuat subfolder di sana yang disebut Notebook atau sesuatu yang lain yang mudah diingat.

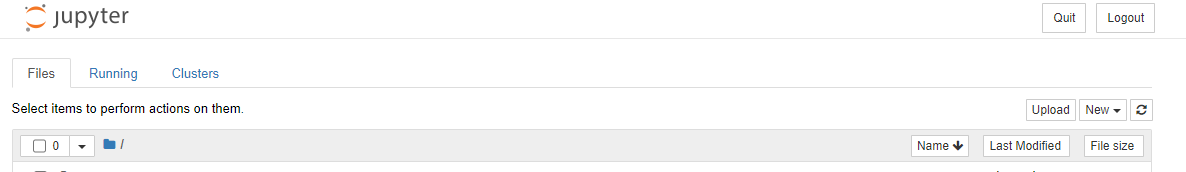
Kemudian pergi saja ke lokasi itu di terminal Anda dan jalankan perintah berikut:



Ini akan memulai Jupyter dan browser default Anda akan mulai (atau membuka tab baru) ke

following URL: <http://localhost:8888/tree>

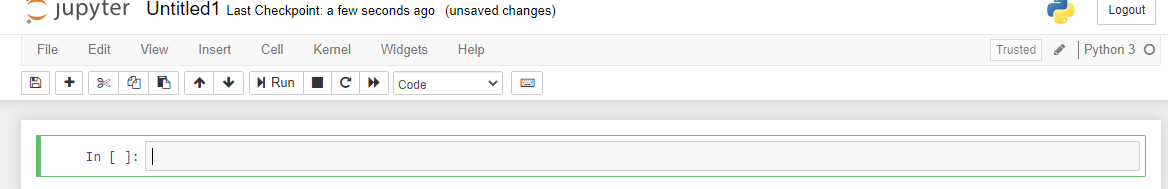
**This will start up Jupyter and your default browser should start (or open aYour browser should now look something like this:a new tab) to the**

****

## **Creating a Notebook**

Sekarang setelah Anda tahu cara memulai server Notebook, Anda mungkin harus belajar cara membuat dokumen Notebook yang sebenarnya.

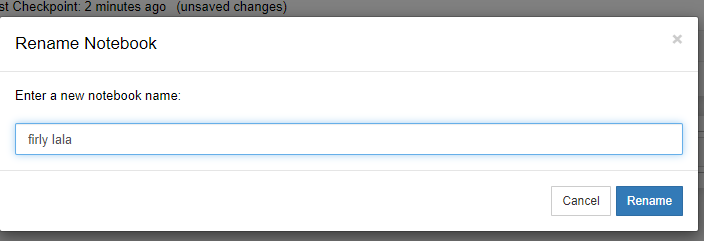
Yang perlu Anda lakukan adalah mengklik tombol Baru (kanan atas), dan itu akan membuka daftar pilihan. Di mesin saya, saya kebetulan memasang Python 2 dan Python 3, jadi saya bisa membuat Notebook yang menggunakan keduanya. Demi kesederhanaan, mari kita pilih Python 3.



**Naming**

Anda akan melihat bahwa di bagian atas halaman adalah kata Untitled. Ini adalah judul untuk halaman dan nama Notebook Anda. Karena itu bukan nama yang sangat deskriptif, mari kita ubah!

Gerakkan mouse Anda ke kata Untitled dan klik pada teks. Anda sekarang akan melihat dialog dalam browser berjudul Rename Notebook. Mari kita ganti nama ini menjadi nama anda:



**REFERENCES**

[1] TIOBE Software Index (2011). “TIOBE Programming Community Index Python”.

[2] “Programming Language Trends- O’Reilly Radar”. Radar.oreilly.com . 2 August 2006

[3] “The RedMonk Programming Language Rankings: January 2011 – tecosystems” Redmonk.com

[4] Summerfield, Mark. Rapid GUI Programming with Python and Qt

[5] Kuhlman, Dave. "A Python Book: Beginning Python,Advanced Python, and Python Exercises".

[6] <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0085777>

[7] MODEL PEMILIHAN SERVER SIDE SCRIPTING UNTUK SISTEM INFORMASI SEKOLAH: STUDI KOMPARASI ASP, JSP, PHP, PYTHON, DAN RUBY .

[8]