

クラウドファンディングにおける成功の判別分析

プロジェクトマネジメントコース

ソフトウェア開発管理グループ

矢吹研究室

1242105

三浦泰介

目次

第 1 章	序論	5
1.1	本章の構成	5
1.2	研究の背景	5
1.3	研究目的	5
1.4	研究方法	6
1.5	プロジェクトマネジメントとの関連性	6
第 2 章	クラウドファンディングについて	7
2.1	本章の構成	7
2.2	クラウドファンディングとは	7
2.3	クラウドファンディングの種類	7
2.4	クラウドファンディングの歴史	7
第 3 章	データマイニングについて	9
3.1	本章の構成	9
3.2	データマイニングとは	9
3.3	データマイニングの事例	9
3.4	データマイニングのツール	9
第 4 章	R 言語について	11
4.1	本章の構成	11
4.2	R 言語とは	11
4.3	R の導入	11
4.4	R の起動と終了	12
第 5 章	クローラーについて	15
5.1	本章の構成	15
5.2	クローラーとは	15
5.3	wget とは	15
5.4	wget オプション	15
第 6 章	クローラーの導入と運用	21
6.1	本章の構成	21
6.2	Ubuntu について	21
6.3	Ubuntu の導入	22
6.4	クローラーの導入	26

参考文献

31

第 1 章

序論

1.1 本章の構成

本章では、本研究の背景・目的・方法・プロジェクトマネジメントとの関係性を記す。

1.2 研究の背景

クラウドファンディングと呼ばれるインターネットを用いた資金調達の手法が、近年、世界中ではやりを見せしており、日本においてもクラウドファンディングを利用して資金調達を行う、プロジェクトが数多く出てきている。クラウドファンディングは誰でも行う権利があり、プロジェクトの規模の大きさも問わないため多くのアーティストやベンチャー企業などから注目を集めており、数多くのいままでになかった製品やサービスが実現化している。プロジェクトにおいて資金集めは極めて重要であり、当然、資金集めができなければプロジェクトは破綻してしまうため、クラウドファンディングを利用する場合は、まずはプロジェクトの成功させる前にクラウドファンディングという一つのプロジェクトを成功させなければならない。クラウドファンディングの成功の有無に関わる成功要因を明らかにし、クラウドファンディングを行う人や出資する人の参考になる情報を見つけられないか考えた。

現在、クラウドファンディングは資金提供者に対するリターンの形で3つの種類に分類分けをされている。

寄付型 金銭的リターンのない

投資型 金銭的リターンのある

購入型 権利や物品を購入することで支援する

日本では金融商品取引法が2014年に改正されるまで、法規制の問題から、見返りを得ない寄付型、購入型に限られていた。購入型はリターンがあるためリターンを目当てに多くの出資者が集まる傾向があり、購入型は日本で一番市場が大きいクラウドファンディングの形態である。今回は市場が、最も大きい購入型のプロジェクトの成功率に関わる成功要因を探すこととする。

1.3 研究目的

クラウドファンディングサイトに掲載されているプロジェクトから、支援の金額、コースの数など支援者側から見える情報をデータとして集め、決定木を描くことで、クラウドファンディングにおける成功要因を明らかにし、クラウドファンディングを用いたプロジェクトを行う際や投資する際の参考となる指標を作ることを目的とする。

1.4 研究方法

監視するクラウドファンディングサイトの選定を行い，毎日定時にデータ収集を行う．プロジェクトの内容や，代表者から知り得る情報を元にデータマイニングの手法の一つである決定木分析を用いて，成功の成否に繋がるパターンを見つけ出す．また成功予測を行い，実際の成否と一致率を上げることで憑性を確保する．

1.5 プロジェクトマネジメントとの関連性

クラウドファンディングを成功させるまでを一つのプロジェクトと考え，その成功率をあげることでクラウドファンディングで資金調達を行うプロジェクトの成功率をあげることに繋がると考える．

第 2 章

クラウドファンディングについて

2.1 本章の構成

本章では、研究対象であるクラウドファンディングについて解説を記す。

2.2 クラウドファンディングとは

クラウドファンディングとは活動資金が必要な組織や団体に不特定多数の人がインターネットを經由し、資金提供を行うことを指す。群衆（crowd）資金調達（funding）を組み合わせた造語であり、ソーシャルファンディングと呼ばれることもある。現在は、クラウドファンディングの参加者を集めるサイトがありそこにプロジェクトを投稿し、出資者を募る形が多く、目標達成金額の一部をサイトに支払う場合が多い。

2.3 クラウドファンディングの種類

クラウドファンディングは資金提供者に対するリターンの形で 3 つの種類に分類分けがされている。

寄付型 金銭的リターンのない

投資型 金銭的リターンのある

購入型 権利や物品を購入することで支援する

日本においては購入型が最も認知されており、プロジェクト数も多い。

2.4 クラウドファンディングの歴史

クラウドファンディングの雛形の前身は 17 世紀に活躍した書籍編集者のジョン・テイラー氏が、書籍の印刷代を募るために使われた寄付ビジネスモデルだったと言われている。寄付のリターンが標題紙に名前を掲載する権利を提供することであり、現在でも映画のスタッフロールや出版物に名前が掲載されることはポピュラーなりターンとして行われており、クラウドファンディングと通じるものがあるとされている。

第 3 章

データマイニングについて

3.1 本章の構成

本章では、研究で使用するデータマイニングについて解説を記す。

3.2 データマイニングとは

データマイニングとは、1990 年代中頃から用いられるようになった言語であり、大量のデータから統計学、パターン認識、人工知能などのデータ解析の技法を使い、大量のデータの中から知識を取り出す技術のことである。テキストを対象とするものをテキストマイニング、中でもウェブページを対象にしているものはウェブマイニングと呼ばれている。

3.3 データマイニングの事例

データマイニングの成功事例は、数多く報告されており、有名な話としておむつとビールの話がある。データマイニングにバスケット分析という、POS データなどの取引データを分析する手法があり、「おむつを買った人はビールを買う傾向がある」という分析結果が 1990 代半ばから 2000 年台初めにかけてメディアや講演などでよく語られ、データマイニングという言葉と概念を有名にした。明確な答えはないが、かさばる紙おむつを父親に買いに行かせた時に、父親がついでにビールを買っていく推測と、深夜に 30 代の男性がおむつと一緒にビールを買っている傾向が発見されたことから、おむつを突然切らしてしまった母親に、父親がおむつの買い物を頼まれそのついでにビールを買っているのではないかという推測の 2 つが挙げられている。おむつとビールの売り場を近くにおいたことで店の売上向上につながった話もあり、データマイニングの結果から推測することで顧客の潜在的なニーズを引き出すことができる。

3.4 データマイニングのツール

大量のデータを扱うには、ツールが必要である。データマイニングのツールでは SAS, SPSS, S 言語が有名である。しかしデータマイニング機能を揃えた SAS, SPSS, S 言語のパッケージは値段が高く、個人ユーザが簡単に使えるものではない、そこで現在普及してきている S 言語並みの機能を持つフリーソフト R を今回はデータマイニングのツールとして用いることとする。

第 4 章

R 言語について

4.1 本章の構成

本章では、研究で使用する R 言語についてと利用方法について解説を記す。

4.2 R 言語とは

ニュージーランドのオークランド大学の統計学科の Ross Ihaka と、アメリカのハーバード大学生物統計学科の Robert Gentleman によって開発が始められ、1997 年からは多くの賛同者が加わり、開発が続けられているオープンソース方式のデータ解析・処理の専用ソフトである。[1] R 言語の名前の由来は統計解析言語である S 言語の独自実装であるので、S 言語の「一歩手前」の「R」という意味や、創設者二人の頭文字である R に由来しているという説がある。[2] R 言語の特徴としては、汎用的な言語とはちがひ、解析に特化した言語であるということ、またオープンソースであるため必要に応じて拡張機能を手に入れてくることで様々な用途に対応できることや、簡単なコマンドにより様々な機能が実現できることがあげられる。

4.3 R の導入

R 言語は UNIX, Windows, Mac など様々な OS で使用することが可能である。今回は、Windows 版の導入方法を記す。R 言語の公式サイトに <https://www.r-project.org/> アクセスし、Download の項目にある CRAN をクリックする。

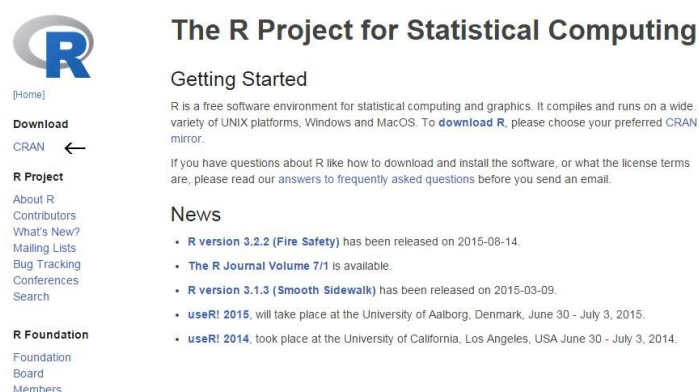


図 4.1 R インストール手順

「JAPAN」のリンクを探し、統計数理研究所か山形大学の R のミラーサイトをクリックする。

Iran	http://cran.ac.ir/	Ferdowsi University of Mashhad
Ireland	http://ba.hawaii.edu/mirrors/cran-cranproject.org/	HEAnetDublin
Italy	http://cran.mirror.net/mirrors/CRAN/	Garr Mirror, Milano
	http://cran.stat.uned.it/	University of Padua
	http://cran.stat.uned.it/	Università degli Studi di Palermo
Japan	http://cran.kan.ac.jp/	Institute of Statistical Mathematics, Tokyo
	http://ba.yz.yamagata-u.ac.jp/pub/cran/	Yamagata University
Korea	http://cran.nyu.ac.kr/	NyuR Corporation, Seoul
	http://hwa.hwa.ac.kr/cran/	Graduate School of Public Health, Seoul National University, Seoul
	http://cran.hawaii.edu/	The Genome Institute of UNIST (Ulsan National Institute of Science and Technology)
Lebanon	http://mirror.lau.edu.lb/	Lebanese American University, Byblos
Mexico	http://cran.lam.mx/	Instituto Tecnológico Autónomo de México
...	http://www.est.cobos.mx/RR-mirror/	Colegio de Postgraduados, Texcoco

図 4.2 R インストール手順 2

画面上部にある「Download R for Windows」をクリックする。



CRAN
Mirrors
What's new?
Task Views
Search

About R
R Homepage
The R Journal

Software
R Sources
R Binaries
Packages
Other

Documentation
Manuals
FAQs
Contributed

R-3.2.2 for Windows (32/64 bit)

[Download R 3.2.2 for Windows](#) (62 megabytes, 32/64 bit) ←
[Installation and other instructions](#)
[New features in this version](#)

If you want to double-check that the package you have downloaded exactly matches the package distributed by R, you can compare the [md5sum](#) of the .exe to the [true fingerprint](#). You will need a way of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

Frequently asked questions

- [Does R run under my version of Windows?](#)
- [How do I update packages in my previous version of R?](#)
- [Should I run 32-bit or 64-bit R?](#)

Please see the [R FAQ](#) for general information about R and the [R Windows FAQ](#) for Windows-specific information.

Other builds

- Patches to this release are incorporated in the [r-patched snapshot build](#).
- A build of the development version (which will eventually become the next major release of R) available in the [r-devel snapshot build](#).

図 4.3 R インストール手順 2

リンクの一番上にある base をクリックする。「Download R 3.2.2 for Windows」をクリックすると Download が開始する。

4.4 R の起動と終了

デスクトップ上の R のショートカットアイコンをクリックするか、「スタート」→「すべてのプログラム」→「R」のフォルダにある R をクリックすると RGui ウィンドウが開く。基本的にコンソール上で操作を行い、作成されたグラフはグラフィックウィンドウが出てきて表示がされる。R を終了する際には、コンソールに「q()」または「quit()」と入力し Enter キーを押すことで終了できる。

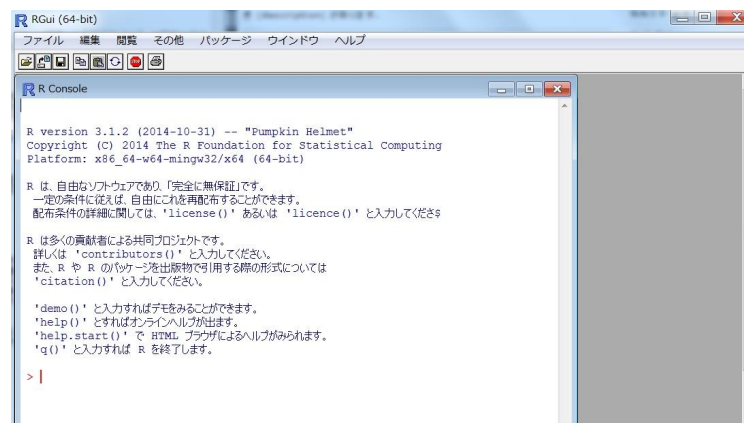


図 4.4 R インストール手順

第 5 章

クローラーについて

5.1 本章の構成

本章では，研究で使ったクローラーの作成方法について記す．

5.2 クローラーとは

ネット上の文章や画像などを周期的に所得し，自動的に保存するプログラムである．「ボット (Bot)」，スパイダー，ロボットとも呼ばれる．今回はフリーソフトウェアである `wget` を利用し，作成したクローラーの説明を行う．

5.3 `wget` とは

`wget` とはウェブサーバーからコンテンツを取得するプログラムであり，多くの UNIX ユーザーから利用されている．また多くの環境で利用可能であり，低速で不安定なインターネットでも確実に動作し，ダウンロードを行えるように設計されていることが特徴である．もしダウンロードが停止した場合，停止した箇所からダウンロードを継続するように試みてファイル全体が取得されるまで繰り返す．また再帰的にダウンロードを行うことが可能であるため，クローラーとして動作させることも可能であり，オプションを組みわせることでウェブサイトから画像や文章をダウンロードすることができる．文法は「`wget` -オプション (オプション引数) URL」のように記述する．

5.4 `wget` オプション

`wget` を使いこなす上で重要であるオプションを解説していく．オプションは大文字と小文字で別の動作をするものがあり，使いたいオプションは大文字か小文字かきちんと確認しておくことが重要である．短縮形が無いオプションに関しては長いオプションで記述する．

表 5.1 スタートアップ

コマンド名	解説
-V	<code>wget</code> のバージョンを表示して終了
-h	オプション一覧のヘルプ
-b	スタート後にバックグラウンドに移行する．
-e	「.wgetrc 形式のコマンドを実行する．」

表 5.2 ログと入力ファイル

コマンド名	解説
-o	ログを FILE に出力する
-a	メッセージを FILE に追記する
-d	デバック情報を表示する.
-q	何も表示しない
-v	冗長な出力をする (デフォルト)
-nv	冗長ではなくする
-report-speed=TYPE	帯域幅を TYPE で出力します.
-i	FILE の中に指定された URL をダウンロードする
-F	入力されたファイルを HTML として扱う
-B	HTML で入力されたファイル (-i -F) のリンクを設定した URL の相対 URL として扱う
-configFILE	設定ファイルを指定する

表 5.3 ダウンロード

コマンド名	解説
-t	リトライ回数の上限を設定 (0 は無制限)
-O	接続を拒否されてもリトライする
-nc	FILE に文書を書き込む
-c	部分的にダウンロードしたファイルの続きから書き始める
-N	ローカルにあるファイルよりも新しいファイルだけ取得する
-no-use-server-timestamps	ローカル側のファイルスタンプにサーバーのものを使わない
-S	サーバーの応答を表示する
-T	全てのタイムアウトを SECONDS 秒に設定する
-dns-timeout=SECS	DNS 問い合わせのタイムアウトを SECS 秒に設定する
-connect-timeout=SECS	接続タイムアウトを SECS 秒に設定する
-read-timeout=SECS	読み込みタイムアウトを SECS 秒に設定する
-w	ダウンロード毎に SECONDS 秒待つ
-waitretry	リトライ毎に 1~SECONDS 秒待つ
-no-proxy	プロキシを使わない
-Q	ダウンロードするバイト数の上限を指定する
-bind-address=ADDRESS	ローカルアドレスとして ADDRESS(ホスト名か IP) を使う
-limit-rate=RATE	ダウンロード速度を RATE に制限する
-no-dns-cache	DNS の問い合わせ結果をキャッシュしない
-restrict-file-names=OS	OS が許しているファイル名に制限する
-ignore-case	ファイル名, ディレクトリ名の比較で大文字小文字を無視する
-4	IPv4 だけを使う
-6	IPv6 だけを使う
-prefer-family=FAMILY	指定したファミリ (IPv6,IPv4,none) で最初に接続する
-user=USER	ftp,http のユーザー名を指定する
-password=PASSWORD	ftp,http のパスワードを指定する
-ask-password	パスワードを別途入力する
-no-iri	IRI サポートを使わない
-local-encoding=ENC	指定した ENC を IRI のローカルエンコーディングにする
-remote-encoding=ENC	指定した ENC をデフォルトのリモートエンコーディングにする
-unlink	上書きする前にファイルを削除する

表 5.4 ディレクトリ

コマンド名	解説
-nd	ディレクトリを作らない
-x	ディレクトリを強制的に作る
-nH	ホスト名のディレクトリを作らない
-protocol-directories	プロトコル名のディレクトリを作る
-P	ファイルを PREFIX 以下に保存する
-cut-dirs=NUMBER	リモートディレクトリ名の NUMBER 階層分を無視する

表 5.5 HTTP オプション

コマンド名	解説
-http-user=USER	http ユーザ名として USER を使う
-http-password=PASS	http パスワードとして PASS を使う
-no-cache	サーバがキャッシュしたデータを許可しない
-default-page=NAME	デフォルトのページ名を NAME に変更します.
-E	HTML/CSS 文書は適切な拡張子で保存する
-ignore-length	Content-Length ヘッダを無視する
-header=STRING	送信するヘッダに STRING を追加する
-max-redirect	ページで許可する最大転送回数
-proxy-user=USER	プロキシユーザ名として USER を使う
-proxy-password=PASS	プロキシパスワードとして PASS を使う
-referer=URL	Referer を URL に設定する
-save-headers	HTTP のヘッダをファイルに保存する
-U	User-Agent として Wget/VERSION ではなく AGENT を使う
-no-http-keep-alive	HTTP の keep-alive (持続的接続) 機能を使わない
-no-cookies	クッキーを使わない
-load-cookies=FILE	クッキーを FILE から読みこむ
-save-cookies=FILE	クッキーを FILE に保存する
-keep-session-cookies	セッションだけで用いるクッキーを保持する
-post-data=STRING	POST メソッドを用いて STRING を送信する
-post-file=FILE	POST メソッドを用いて FILE の中味を送信する
-content-disposition	Content-Disposition ヘッダがあればローカルのファイル名として用いる
-auth-no-challenge	サーバからのチャレンジを待たずに、Basic 認証の情報を送信します。

表 5.6 HTTPS (SSL/TLS) オプション

コマンド名	解説
-secure-protocol=PR	セキュアプロトコルを選択する (auto, SSLv2, SSLv3, TLSv1)
-no-check-certificate	サーバ証明書を検証しない
-certificate=FILE	クライアント証明書として FILE を使う
-certificate-type=TYPE	クライアント証明書の種類を TYPE (PEM, DER) に設定する
-private-key=FILE	秘密鍵として FILE を使う
-private-key-type=TYPE	秘密鍵の種類を TYPE (PEM, DER) に設定する
-ca-certificate=FILE	CA 証明書として FILE を使う
-ca-directory=DIR	CA のハッシュリストが保持されているディレクトリを指定する
-random-file=FILE	SSL PRNG の初期化データに使うファイルを指定する
-egd-file=FILE	EGD ソケットとして FILE を使う

表 5.7 FTP オプション

コマンド名	解説
<code>-ftp-user=USER</code>	ftp ユーザとして USER を使う
<code>-ftp-password=PASS</code>	ftp パスワードとして PASS を使う
<code>-no-remove-listing</code>	.listing ファイルを削除しない
<code>-no-glob</code>	FTP ファイル名のグロブを無効にする
<code>-no-passive-ftp</code>	passive 転送モードを使わない
<code>-retr-symlinks</code>	再帰取得中に、シンボリックリンクでリンクされた先のファイルを取得する

表 5.8 再起ダウンロード

コマンド名	解説
<code>-r</code>	再帰ダウンロードを行う
<code>-l</code>	再帰時の階層の最大の深さを NUMBER に設定する (0 で無制限)
<code>-delete-after</code>	ダウンロード終了後、ダウンロードしたファイルを削除する
<code>-k</code>	HTML や CSS 中のリンクをローカルを指すように変更する
<code>-K</code>	リンク変換前のファイルを .orig として保存する
<code>-m</code>	<code>-N -r -l 0 -no-remove-listing</code> の省略形
<code>-p</code>	HTML を表示するのに必要な全ての画像等も取得する
<code>-strict-comments</code>	HTML 中のコメントの処理を厳密にする

表 5.9 再起ダウンロード時のフィルタ

コマンド名	解説
<code>-A</code>	ダウンロードする拡張子をコンマ区切りで指定する
<code>-R</code>	ダウンロードしない拡張子をコンマ区切りで指定する
<code>-D</code>	ダウンロードするドメインをコンマ区切りで指定する
<code>-exclude-domains=LIST</code>	ダウンロードしないドメインをコンマ区切りで指定する
<code>-follow-ftp=HTML</code>	文書中の FTP リンクも取得対象にする
<code>-follow-tags=LIST</code>	取得対象にするタグ名をコンマ区切りで指定する
<code>-ignore-tags=LIST</code>	取得対象にしないタグ名をコンマ区切りで指定する
<code>-H</code>	再帰中に別のホストもダウンロード対象にする
<code>-L</code>	相対リンクだけ取得対象にする
<code>-I</code>	取得対象にするディレクトリを指定する
<code>-X</code>	取得対象にしないディレクトリを指定する
<code>-np</code>	親ディレクトリを取得対象にしない

第 6 章

クローラーの導入と運用

6.1 本章の構成

本章ではクローラーの調査環境として利用する Ubuntu とクローラーの運用について解説を記す。

6.2 Ubuntu について

Ubuntu（ウブントゥ）とは、コミュニティにより開発されているオペレーティングシステムです。ラップトップ、デスクトップ、そしてサーバーに利用することができます。Ubuntu には、家庭・学校・職場で必要とされるワープロやメールソフトから、サーバーソフトウェアやプログラミングツールまで、あらゆるソフトウェアが含まれています。

Ubuntu は現在、そして将来に渡って無償で提供されます。ライセンス料を支払う必要はありません。Ubuntu をダウンロードすれば、友達や家族と、あるいは学校やビジネスに、完全に無料で利用できます。

私たちは、新しいデスクトップおよびサーバーを 6 ヶ月ごとにリリースすることを宣言しています。これにより、オープンソースの世界が提供する最新の優れたアプリケーションを常に利用できるようにしています。

Quantal Quetzal 12.10 2012 年 10 月 18 日 2014 年 4 月 Ubuntu は、セキュリティに配慮して設計されています。デスクトップおよびサーバーの無償セキュリティアップデートが、少なくとも 9 ヶ月間に渡って提供されます。長期サポート (LTS) 版を利用すれば、5 年間に渡りセキュリティアップデート提供されます。もちろん、LTS 版を利用するために追加の費用は必要ありません。すべての人が無償という同じ条件で、私たちの精一杯の成果を利用することができます。Ubuntu を新しいバージョンにアップグレードする場合も、常に無償です。

Ubuntu のインストールイメージにはデスクトップ環境がひと通り含まれています。さらに、オンラインでソフトウェアを追加することができます。

グラフィカルインストーラにより、素早く簡単にインストールして使い始めることができます。標準的なインストールにかかる時間は 10~20 分未満です。十分に高速な環境であれば、インストールが 5 分程度で終了することもあります。

一度システムをインストールすれば、インターネット、ドローイング、グラフィックス、そしてゲームといったアプリケーションがすぐに使えるようになります。

Ubuntu サーバーでは、ユーザーがセットアップしたものだけが動作し、それ以外はインストールされません。[3]

6.3 Ubuntu の導入

調査環境として利用する Ubuntu の導入を記す。現在 Ubuntu は 15.04 までリリースされているが、利用者が多く、エラーが出た場合でも対応しやすい 12.04 を利用する。また仮想化ソフトの VirtualBox 上で動作させること前提に解説を進める。初めに、Ubuntu の公式サイトにアクセスし、画面中央部の Ubuntu のダウンロードを選択する。



図 6.1 Ubuntu の導入

使用しているコンピューターのメイン OS を Ubuntu に変更するのであれば、「日本語 Remix イメージのダウンロード」今回のように VirtualBox 上で動作させる場合には「日本語 Remix 仮想ハードディスクイメージのダウンロード」を選択する。



図 6.2 Ubuntu の導 2

画面上部の Ubuntu 12.04 LTS の項目内にある、「ubuntu-ja-12.04-desktop-i386-vhd.zip」を選択し、ダウンロードを行う。



図 6.3 Ubuntu の導入 3

VirtualBox 上で新規仮想マシンを作成し、Ubuntu のインストールを進める。自家国語を選択し、ubuntu をインストールを選択する。



図 6.4 Ubuntu の導入 4

次に自分の国を設定する。時刻に関わる設定のため研究を行う国で設定を行う。続けるを選択し、次の画面でキーボードレイアウトを行う。

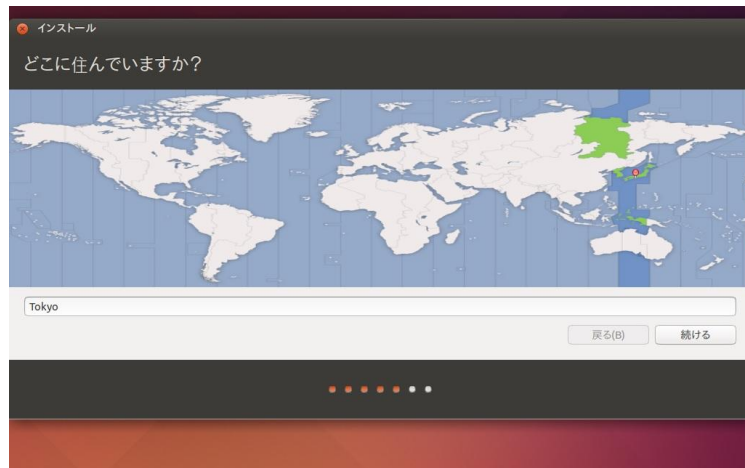


図 6.5 Ubuntu の導入 5

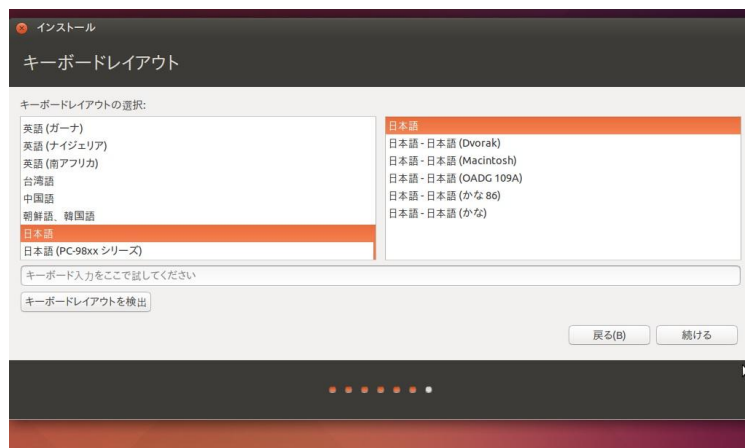


図 6.6 Ubuntu の導入 6

進めるとユーザーの設定とパスワードの設定を行う。ここで使用するパスワードは root 化をする際などでも使用するため、忘れないように保存をしておく。続けるを選択するとインストールが始まる。

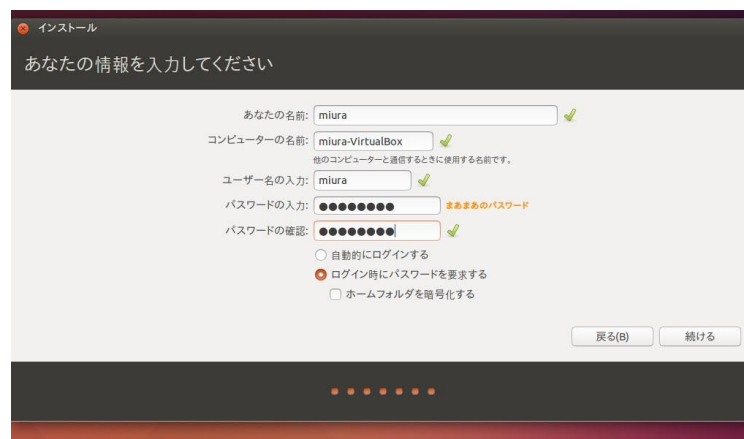


図 6.7 Ubuntu の導入 7

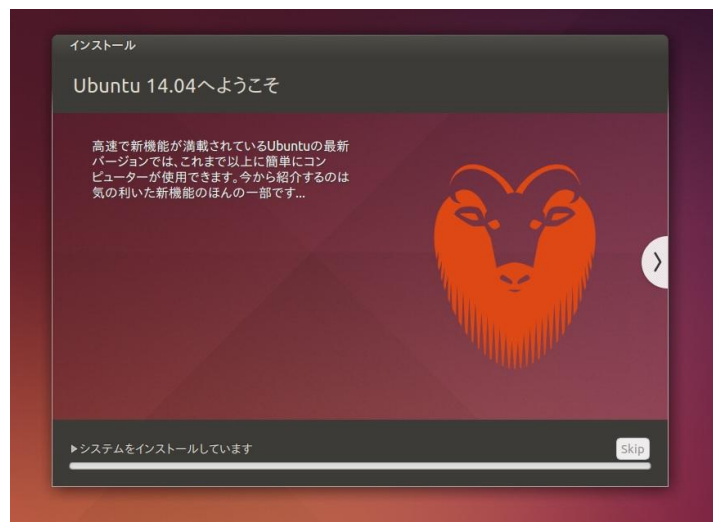


図 6.8 Ubuntu の導入 8

インストール完了が表示されたら再起動を行う。

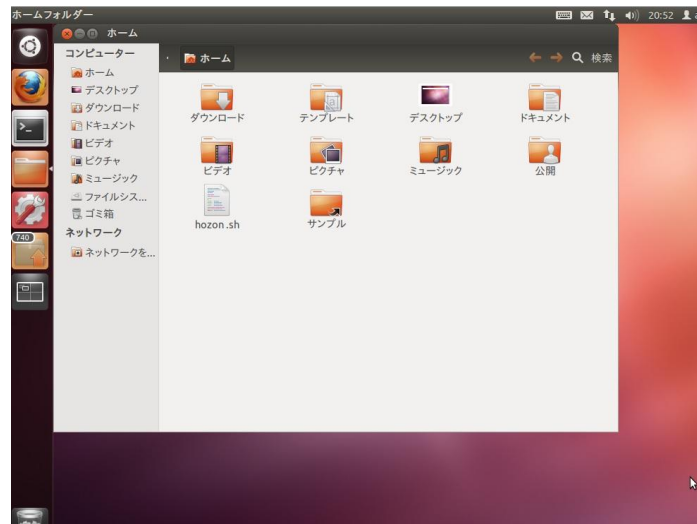


図 6.11 クローラーの導入 2

hazon.sh の内容は以下のとおりである。

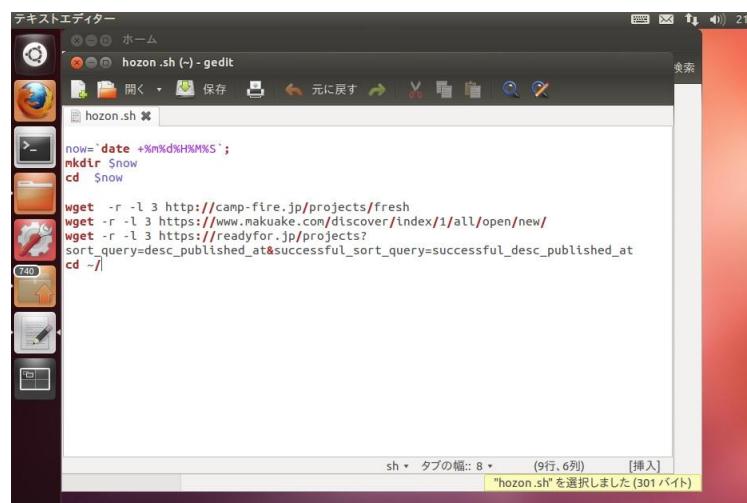


図 6.12 クローラーの導入 3

毎日、同時刻にデータを収集する際に、同じフォルダで保存し続けるのを避けるため、Linux コマンドの `date`, `cd`, `mkdir` を組み合わせてプログラムが動き出した現在時刻をフォルダ名に設定するようにしてある。 `wget` を利用し、各サイトごとに順番にスタート設定した URL から再帰的にファイル入手するように設定してある。以下に今回使用したコマンドの解説を記す。

表 6.1 [date] 日付, 時刻を表示, 設定する

文字	解説
%H	時 (00~23)
%I	時 (01~12)
%k	時 (0~23)
%l	時 (1~12)
%M	分 (00~59)
%p	AM あるいは PM のロケール (国や地域に合わせた文字列)
%r	12 時間形式の時刻 (HH:mm:ss [AP]M)
%s	1970-01-01 00:00:00 UTC からの秒数
%S	秒 (00~61)
%T	24 時間形式の時刻 (HH:mm:ss)
%a	ロケールによる省略形の曜日の名前 (Sun~Sat)
%A	ロケールによる完全に表記した曜日の名前 (Sunday~Saturday)
%b	ロケールによる省略形の月の名前 (Jan~Dec)
%B	ロケールによる完全に表記した月の名前 (January~December)
%c	ロケールによる日付と時刻 (Sat Nov 04 12:02:33 EST 1989)
%d	日 (月内通算日数) (01~31)
%D	日付 (MM/DD/YY)
%j	年内通算日数 (001~366)
%m	月 (01~12)
%w	週のうちの曜日 (0~6) で 0 が日曜日に対応
%x	ロケールによる日付の表現 (MM/DD/YY)
%y	西暦の下 2 けた (00~99)
%Y	年 (1970~)

表 6.2 [cd] ディレクトリを移動する

文字	解説
/	ルート・ディレクトリ
.	現在のディレクトリ
...	親ディレクトリ
~	ホーム・ディレクトリ

表 6.3 [mkdir] ディレクトリを作成する

文字	解説
-m	ディレクトリのモードを設定する
-p	指定したディレクトリをサブディレクトリごと作成する.
-v	ディレクトリを作成する毎にメッセージを出力する
-help	mkdir コマンドの使用方法を表示する
-version	バージョン情報を標準出力に表示する
directory	作成するディレクトリ名を指定する

```
al@al-VirtualBox:~$ crontab -u al -e
Select an editor. To change later, run 'select-editor'.
 1. /bin/ed
 2. /bin/nano      <---- easiest
 3. /usr/bin/vim.tiny
Choose 1-3 [2]:
```

図 6.13 クローラーの導入 4

ホーム上にあるシェルを毎日、同じ時間に自動で動かすために、端末から `crontab` を利用して時間が来たらシェルを呼び出し、プログラムをスタートするように設定する。

- 端末を起動し、`crontab -u ユーザー名 -e` と入力をし、Enter を押す。
- 表示された 3 つの選択肢の中から好きなエディターを選ぶことができる。今回は 2 を選択する。
- 2 を入力し Enter を押すと下記の画面が表示される

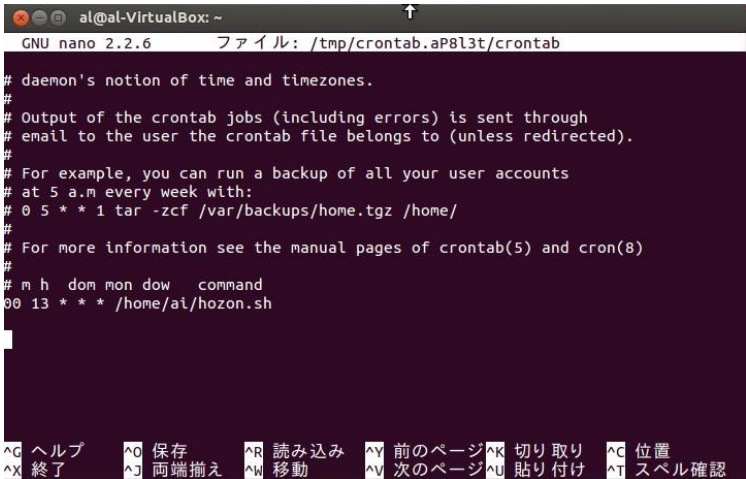


図 6.14 クローラーの導入 5

エディターでプログラムを行う時刻とプログラムを指定する。文法は以下のとおりである。
分 時 日 月 曜日 コマンド

表 6.4 記述方法

文法	解説
分	分を「0～59」で指定する。ワイルドカード (*) を記述すると毎分となる。
時	時間を「0～23」で指定する。ワイルドカード (*) を記述すると毎時となる。
日	日を「1～31」で指定する。ワイルドカード (*) を記述すると毎日となる。
月	月を「1～12」もしくは「jan～dec」で指定する。ワイルドカード (*) を記述すると毎月となる。
曜日	曜日を「0～7」(0,7 は日曜日) もしくは「sun～sat」で指定する。ワイルドカード (*) を記述すると毎日となる。
コマンド	実行したいコマンドやシェルを記述します。

今回のクローラーの稼働設定を例としてあげると、
00 13 * * * /home/al/hozon.sh
毎日 13:00 に/home/al にある hozon.sh を実行する, となる。

参考文献

- [1] 金明哲. R によるデータサイエンス, 2007.
- [2] 山崎明子. 統計解析向けのプログラミング言語「r 言語」の魅力と汎用言語との違い, 2014.
- [3] Ubuntu. Ubuntu とは, 2015.