アドヴァンスト・ソフト・エンジニアリング
夏のインターンシップ

-機械学習による予測を体験してみよう-

2021年9月8日～9月10日

株式会社アドヴァンスト・ソフト・エンジニアリング

Agenda

◇1日目

会社紹介とセキュリティ教育

インターンシップ概要

機械学習概論

使用環境の説明

◇1日目～2日目

サンプルプログラムを用いた回帰分析の概要説明

◇2日目～3日目

予測コンテスト

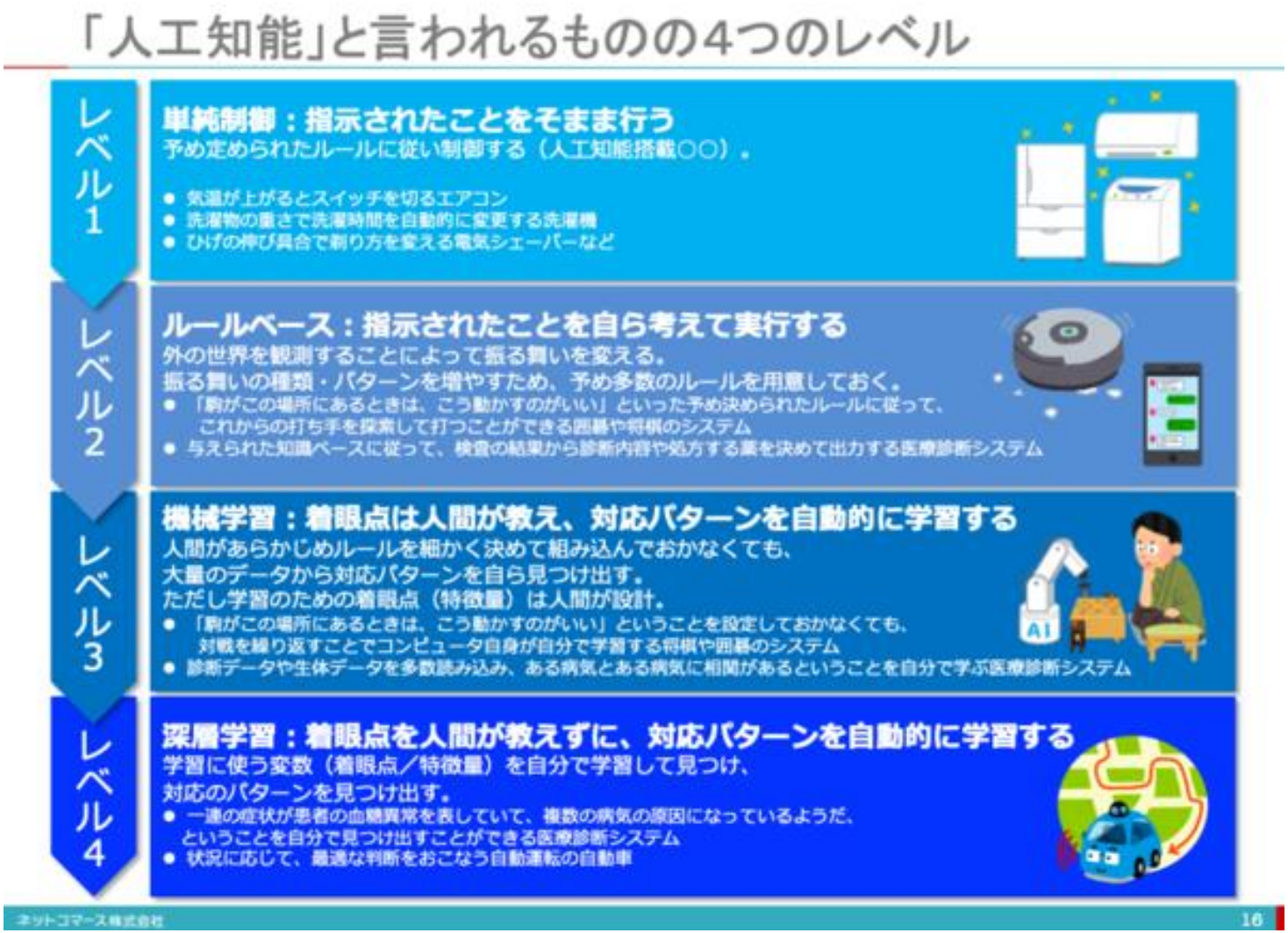
こちらで用意したデータをお渡ししますので、そのデータを用いて実際に予測を行っていただきます。

予測精度の向上を目指して、各自プログラミングやデータの分析を行い、最終的に参加者間での予測精度を競っていただきます。

フィードバック

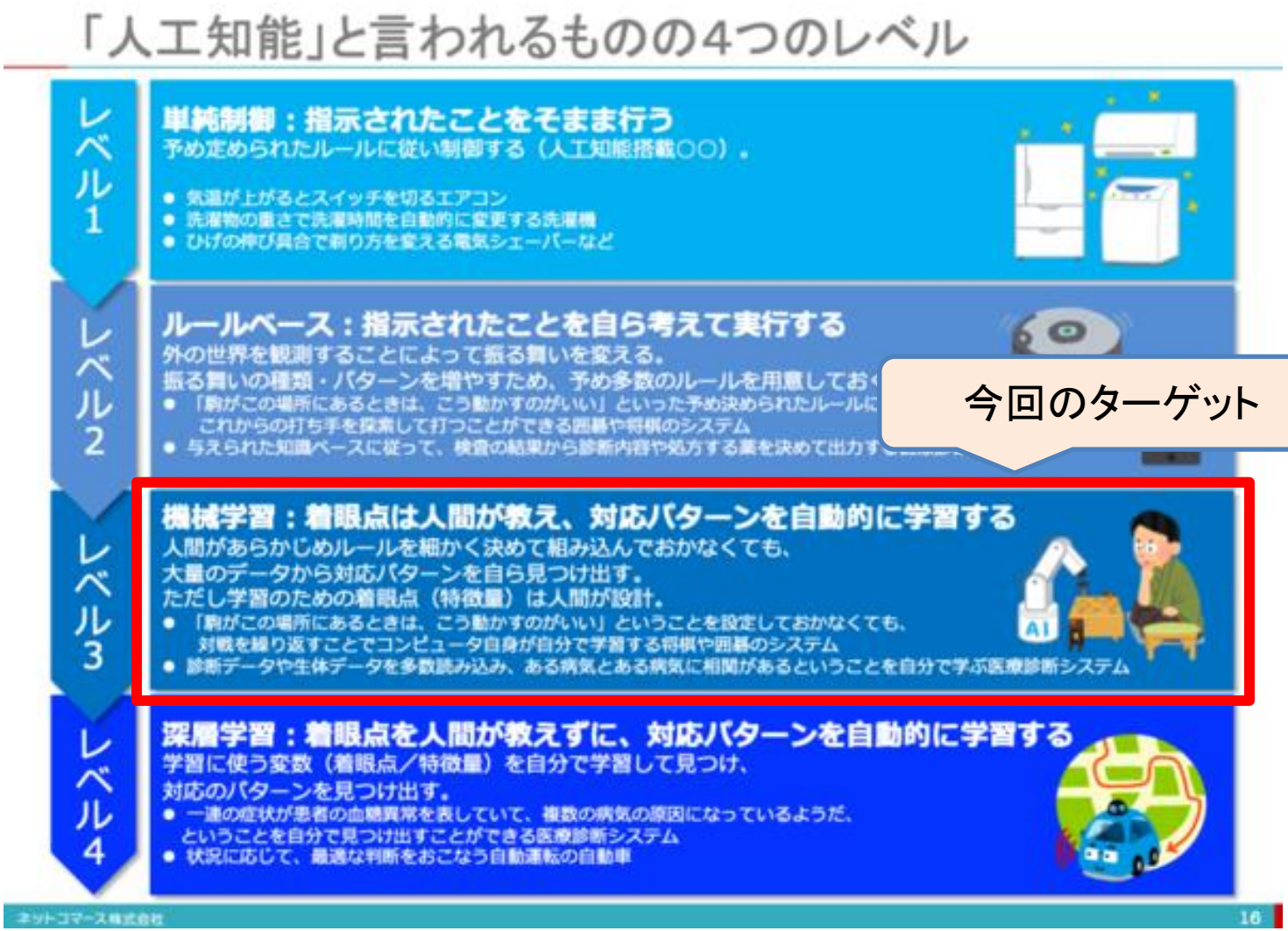
インターンシップ概要
-機械学習概論-

AIとは



※https://blogs.itmedia.co.jp/itsolutionjuku/2020/03/1ai.html

AI(機械学習)とは



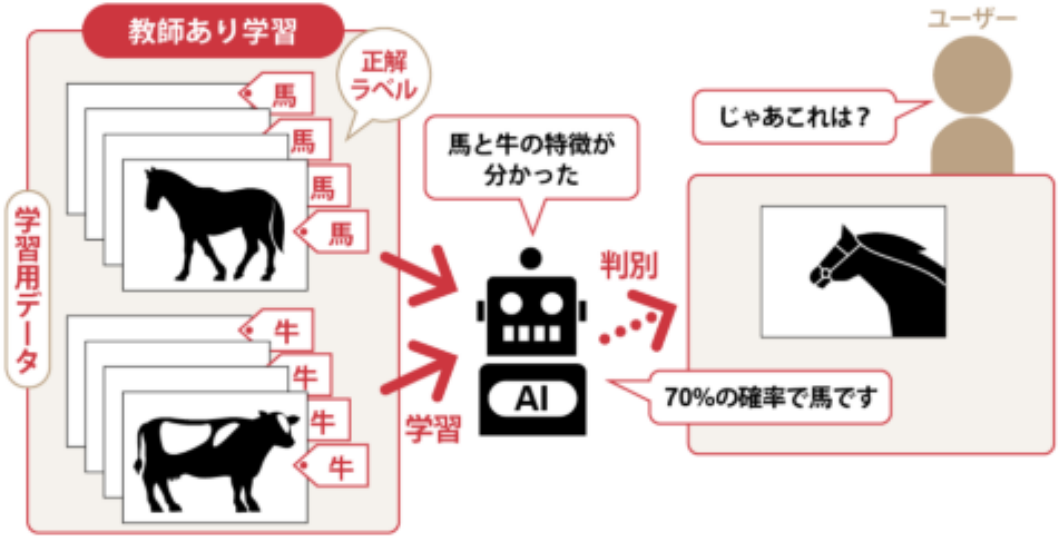
※https://blogs.itmedia.co.jp/itsolutionjuku/2020/03/1ai.html

機械学習とは

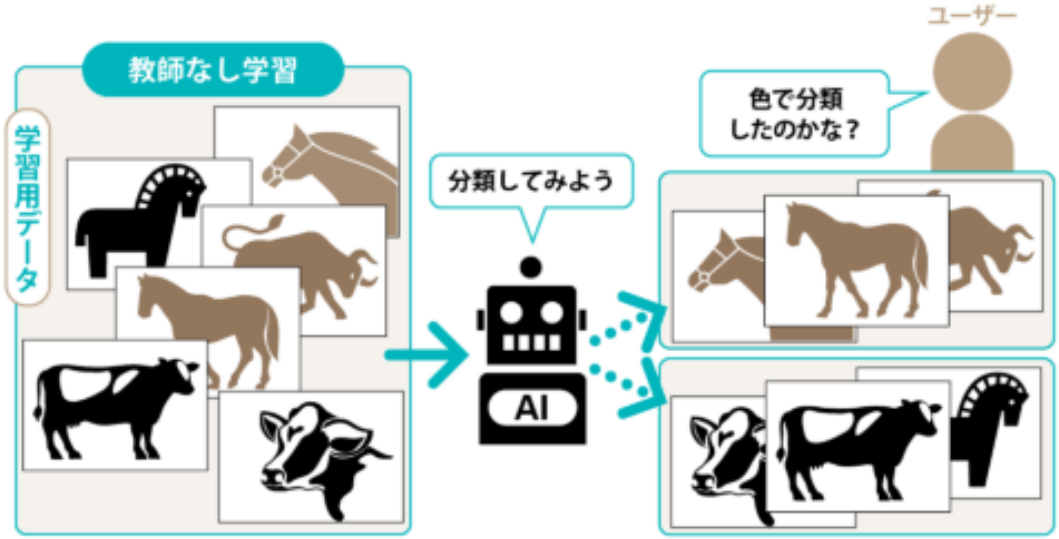
機械に大量のデータからパターンやルールを発見させ、それをさまざまな物事に利用することで判別や予測をする技術

■データからルールやパターンを発見する方法である
■識別と予測が主な使用用途である
■データのどの部分が結果に影響を及ぼしているのか（**特徴量**という）を人間が判断し、調整することで予測や認識の精度を上げる必要がある
■特徴量の設定、学習も自動的に行うのが深層学習(Deep Learning)
→人間が見つけられない特徴を学習可能
→その反面、結果を説明できない(ブラックボックス化)

教師あり学習と教師なし学習



問題と正解がセットになったデータを使って、機械に学習させる。



入力データの中から機械自身が特徴や定義を発見する

https://xtrend.nikkei.com/atcl/contents/18/00163/00004/

回帰分析

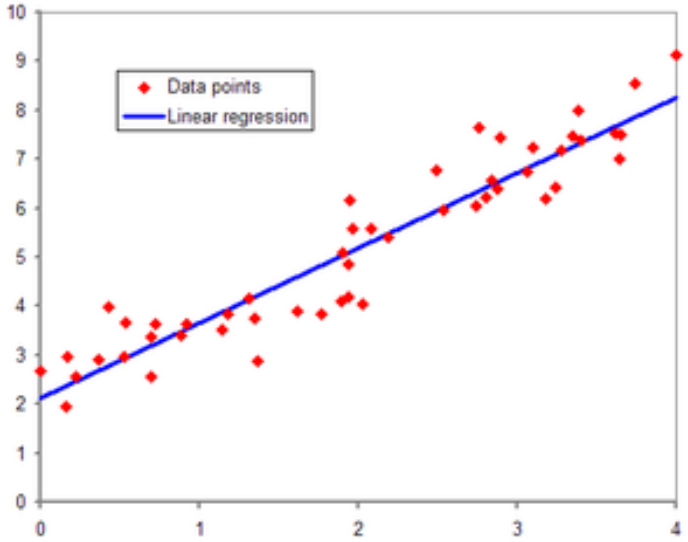
関数をデータに当てはめることによって、ある変数yの変動を別の変数xの変動により説明・予測・影響関係を検討するための手法

■yを目的変数、xを説明変数と呼ぶ

■xが一つの場合を単回帰、複数の場合を重回帰という

単回帰：$y = ax + b$

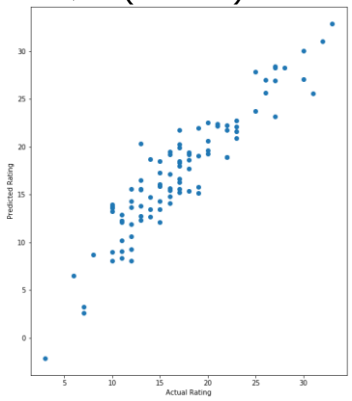
重回帰：$y = a_1x_1 + a_2x_2 + \cdots + a_nx_n + b$



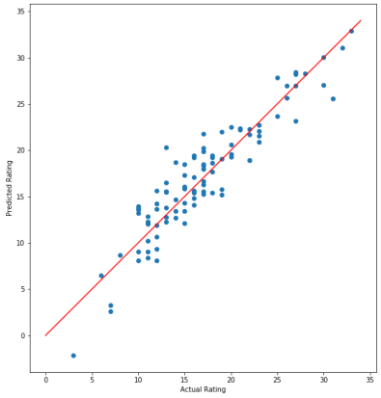
データ(赤い点)に一番フィットする線形関数(青い線)を求める

回帰分析の流れ

教師(学習用)データ

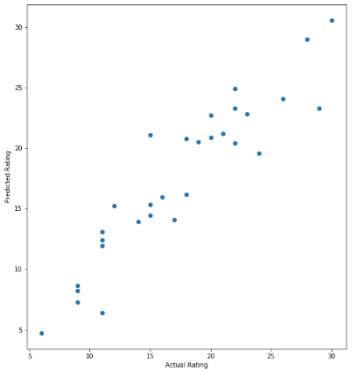


分析

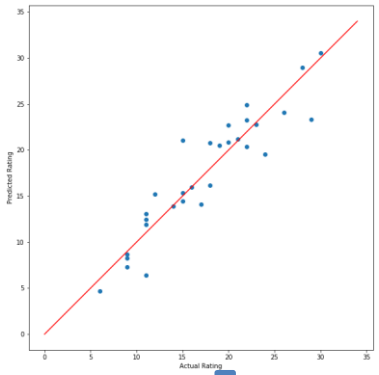


適用

評価(予測用)データ



確認



精度の改善(データ分析)

満足する精度が得られるまで繰り返す

回帰分析を行う際に知っておいた方がいいこと

■係数の意味

$y = a_1x_1 + a_2x_2 + \cdots + a_nx_n + b$

係数(a_1、a_2、・・・、a_n)の(絶対)値が大きい説明変数ほど
目的変数への影響が大きい

■相関

相関：一方の値と、もう一方の値の大きさに関連性があること
一方の値が増えるともう一方も増える：正の相関
一方の値が減るともう一方は増える　：負の相関

相関を数値化した指標が相関係数
相関係数は-1.0～1.0の範囲の値を取る

■因果

原因と結果の間につながりがあること

回帰分析を行う際に知っておいた方がいいこと

■相関と因果

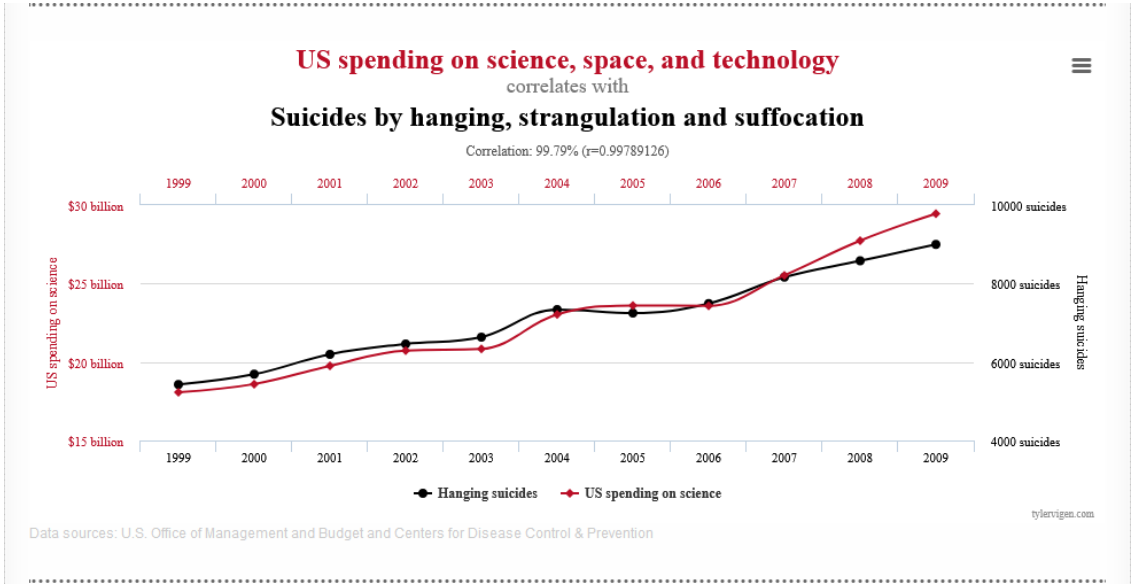
相関あり ≠ 因果あり

　因果を説明できなければ、予測結果の説得力が欠ける

疑似相関

　二つの事象の間に因果関係がないにもかかわらず、あるように見えること。

例）



※https://www.tylervigen.com/spurious-correlations

回帰分析を行う際に知っておいた方がいいこと

■多重共線性

説明変数間の相関係数が高い場合に発生する現象

精度に大きな影響を及ぼす

例)コンビニの月間の売上を予測

目的変数：コンビニエンスストアの月間売上額

説明変数に「雨が降った日数」と「月間の降水量」を入れてしまうと多重共線性が発生する可能性が大きくなる。

※雨が降った日数が多いと、必然的に月間降水量も増加する。

従って、この二つの変数は相関関係が高い。

回帰分析を行う際に知っておいた方がいいこと

■標準化

一般的にデータに含まれる数値には単位や散らばり方が異なるものが混在している。

そこで、標準化という変換を行うことにより異なる性質のデータであっても同じ基準で比較、分析できるようにする。

標準化するための数式

(データ – 平均値) / 標準偏差

標準化することにより平均が0、分散が1のデータとなる。

補足)標準偏差の公式

$$s = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

※標準化を利用した指標が偏差値であり、以下の式で求められる。

(データ – 平均値) / 標準偏差 * 100 + 50

回帰分析を行う際に知っておいた方がいいこと

■データの分割

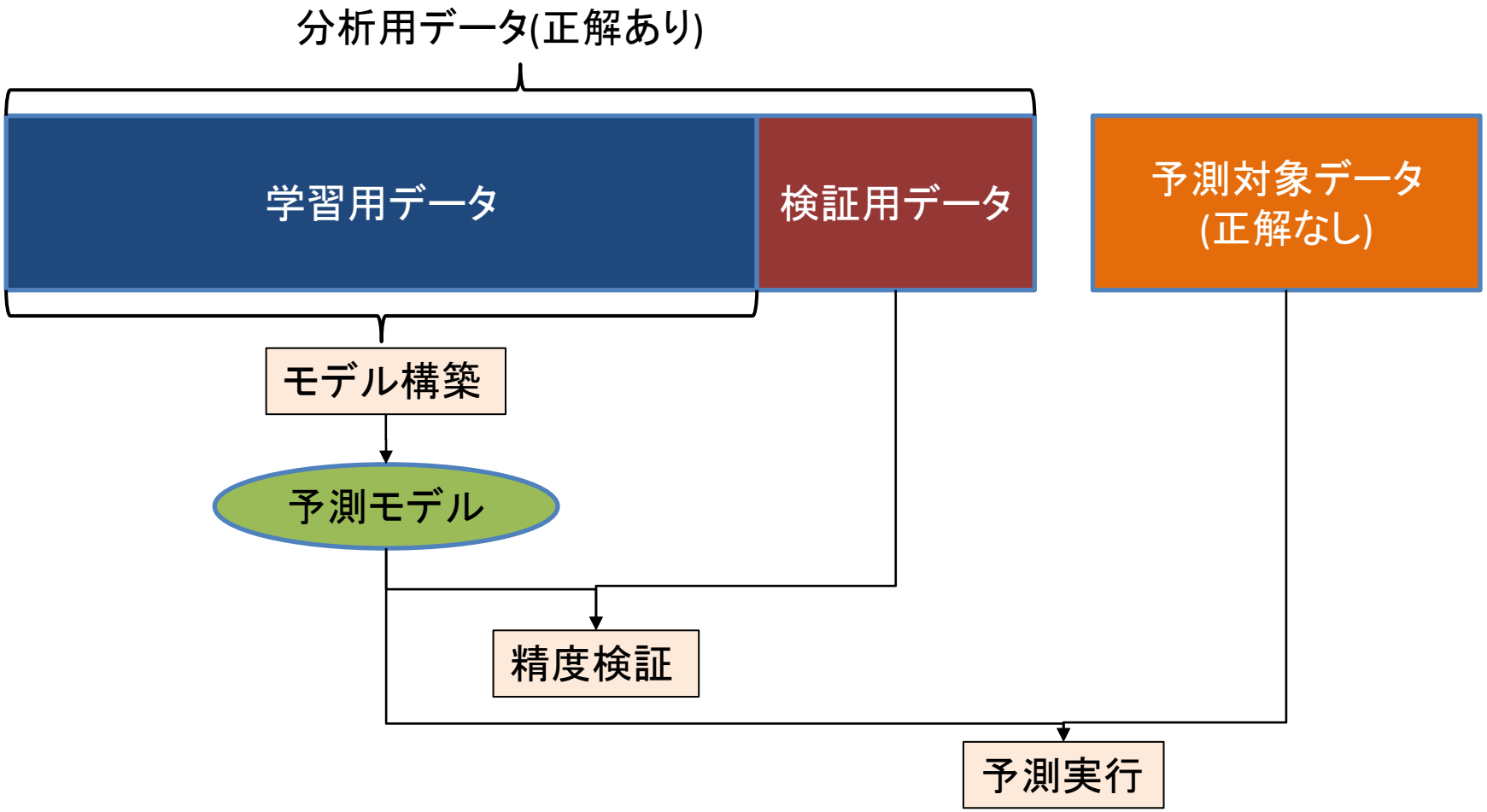
回帰分析を行う場合、一般的には分析用データ(正解がわかっているデータ)を用いてモデルを構築し、予測対象データ(正解がわからないデータ)に対してモデルを適用し、予測結果を得る。

一般的に、モデルの構築時には分析用データを学習用データと検証用データに分割する。
学習用データを用いてモデル構築を行い、そのモデルを検証用データに適用することによって、構築したモデルの予測精度の検証が可能となる。

※予測対象データは正解がわからないので、分析用データの全てを学習用データとしてしまうと、予測精度の検証ができなくなってしまう。

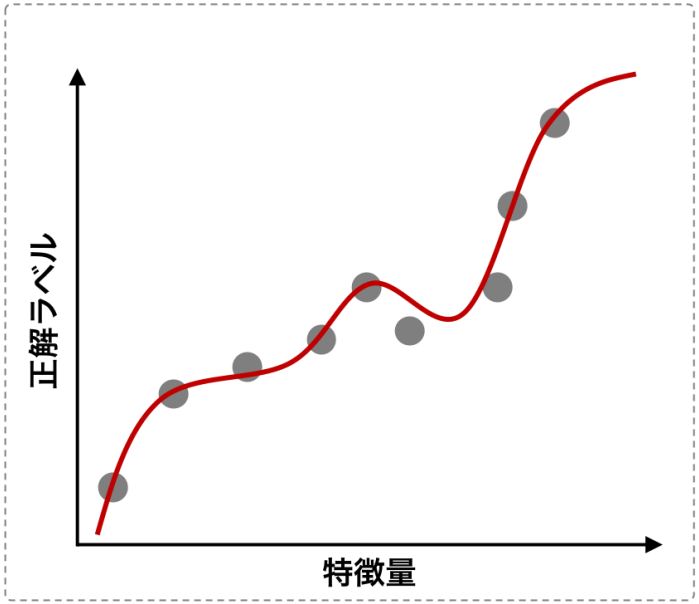
回帰分析を行う際に知っておいた方がいいこと

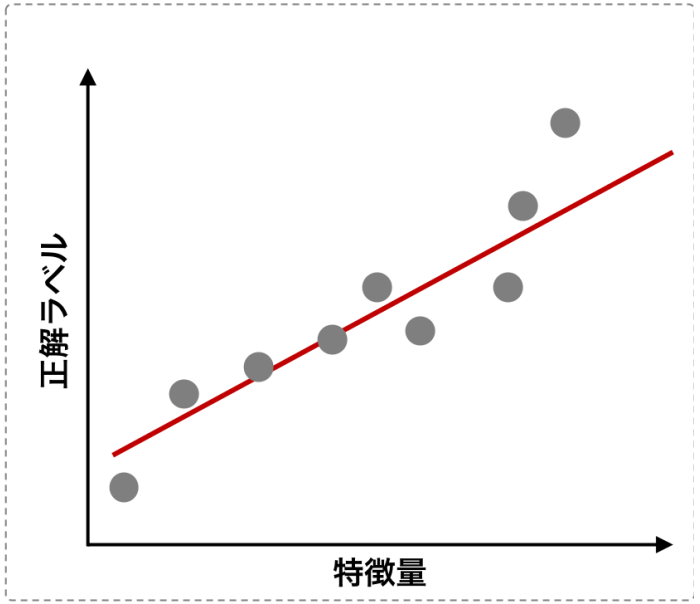
■データの分割(イメージ)



回帰分析を行う際に知っておいた方がいいこと

■過学習





教師データを学習しすぎると、教師データに対してフィットし過ぎたモデルを作成してしまう。その結果、実際の予測データに対する精度が落ちてしまう。

インターンシップ概要
-使用環境の説明-

環境について

■使用する言語

プログラミングの際はpythonを使用します。

機械学習の分野では現在最もポピュラーな言語です。

もちろん機械学習以外の分野でも用いられています。

■使用する開発環境

Jupyter notebookを使用します。

Webブラウザ上で動作する非常にポピュラーな開発環境です。

環境設定に要するコストを抑えることができます。

■その他

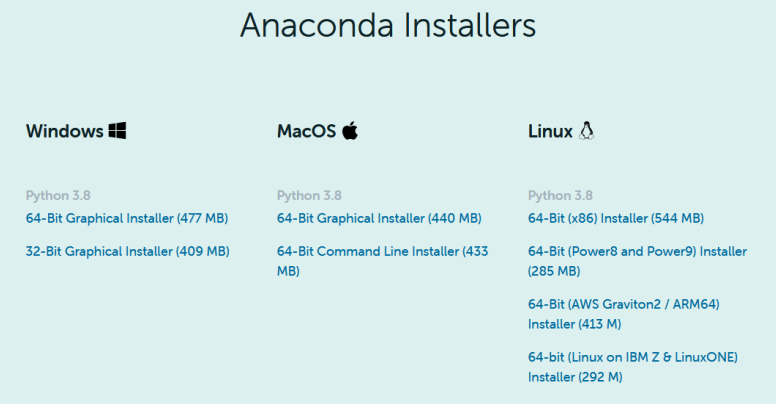
データを概観するためにExcel(または他のスプレッドシートソフトウェア)も使用します。

Notebookのインストール

■既にNotebookの環境をお持ちの方は不要です。

■今回のインターンシップでは”Anaconda”というプラットフォームをインストールしていただきます。
※AnacondaをインストールするとNotebook以外にも色々とインストールされてしまいますが、今回は使用しません。

■以下のURLからインストーラをダウンロードしてください。
https://www.anaconda.com/products/individual
ずーっと下の方にスクロールしていくと以下の様な表示がありますので、お使いのPCに応じたインストーラをダウンロードします。



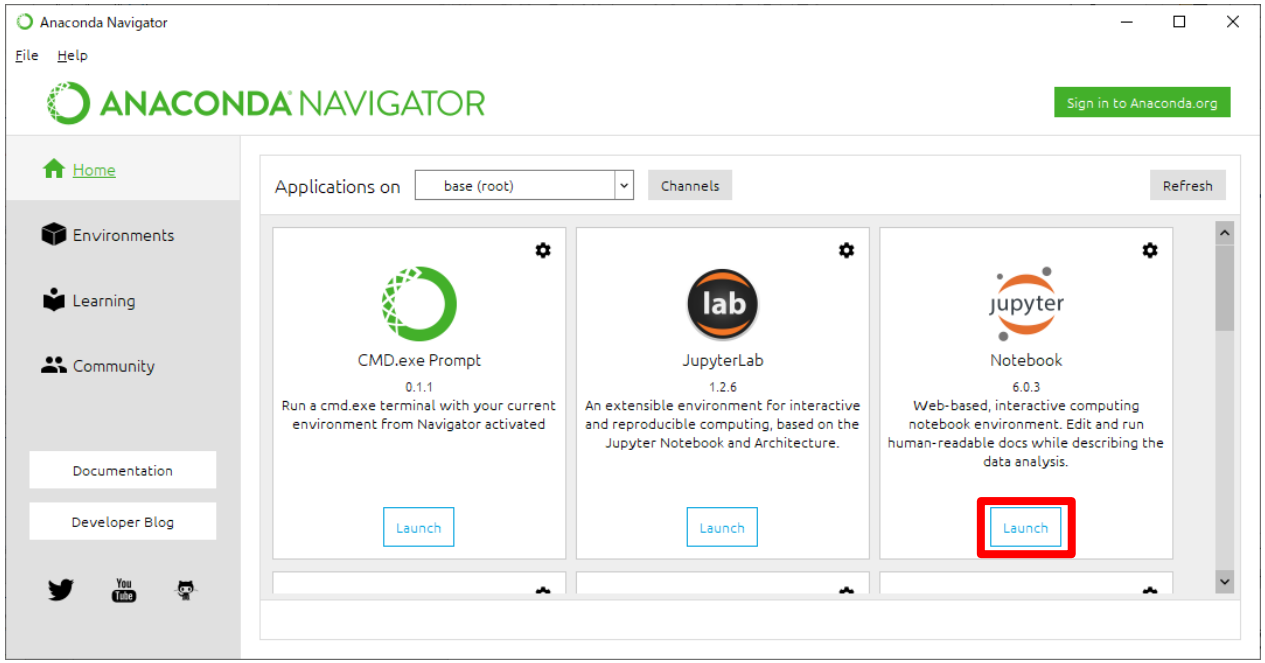
Notebookの起動(Windowsの場合)

■Anacondaのインストールが完了しましたら

スタート→Anaconda3(64-bit)→Anaconda Navigator(anaconda3)

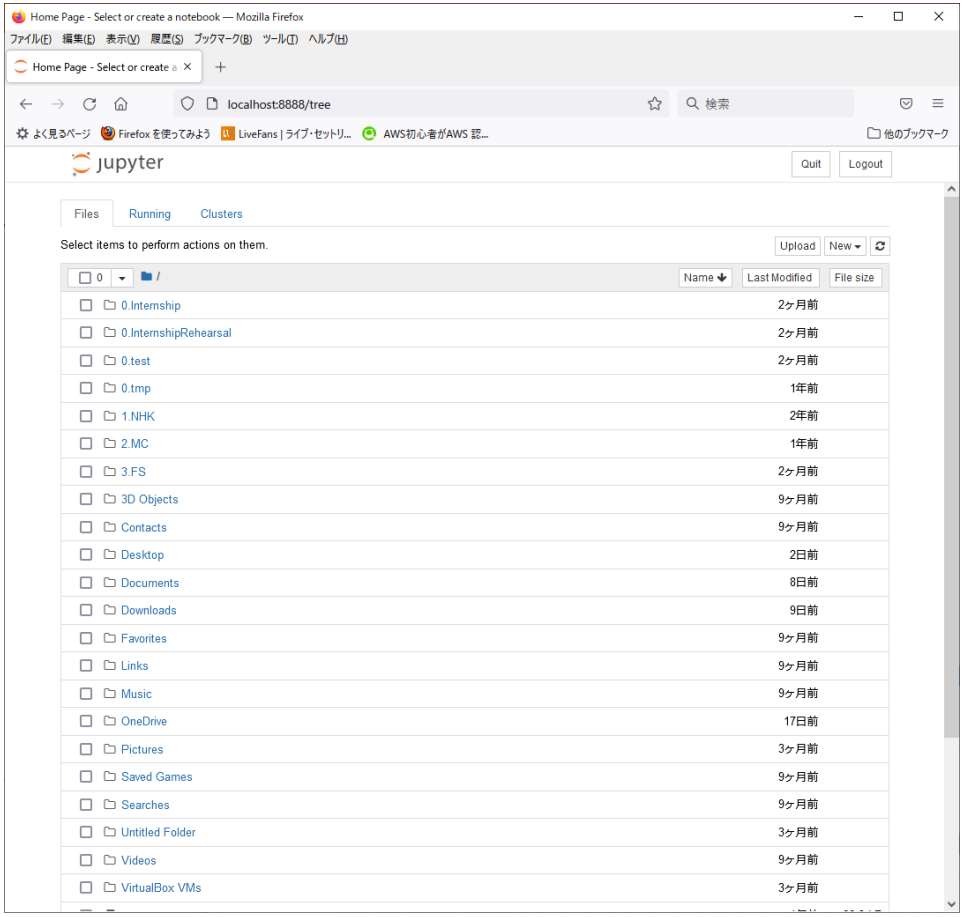
でAnacondaを起動します。

■起動すると以下の様な画面が表示されますので、Jupiter Notebookの“Launch”ボタンをクリックするとNotebookが起動します。



Notebookの起動(Windowsの場合)

■ブラウザが立ち上がりNotebookが起動します。

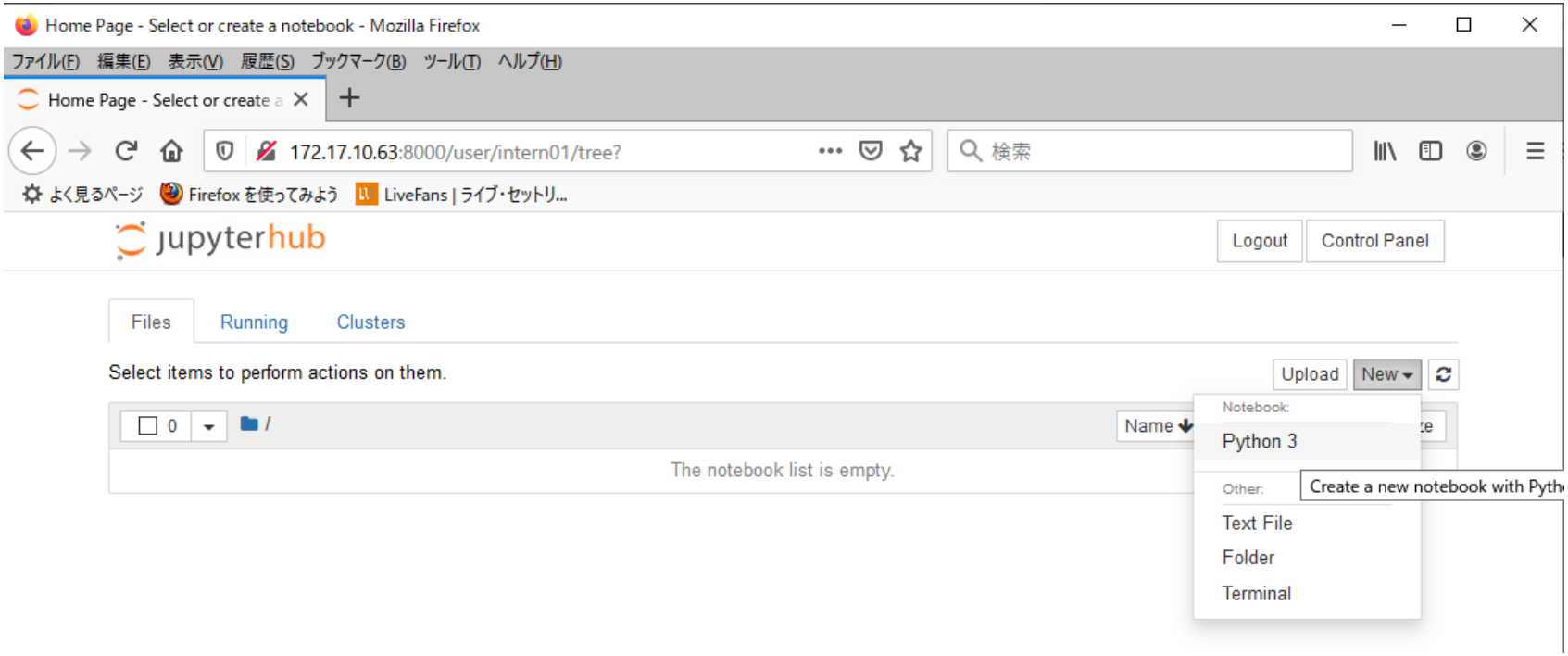


起動時に表示されているフォルダはWindowsのホームディレクトリ(C:¥Users¥xxxxx)です。

インターンシップ用にフォルダを作成するとごちゃごちゃしなくて良いかもしれません。

Notebookの簡単な操作

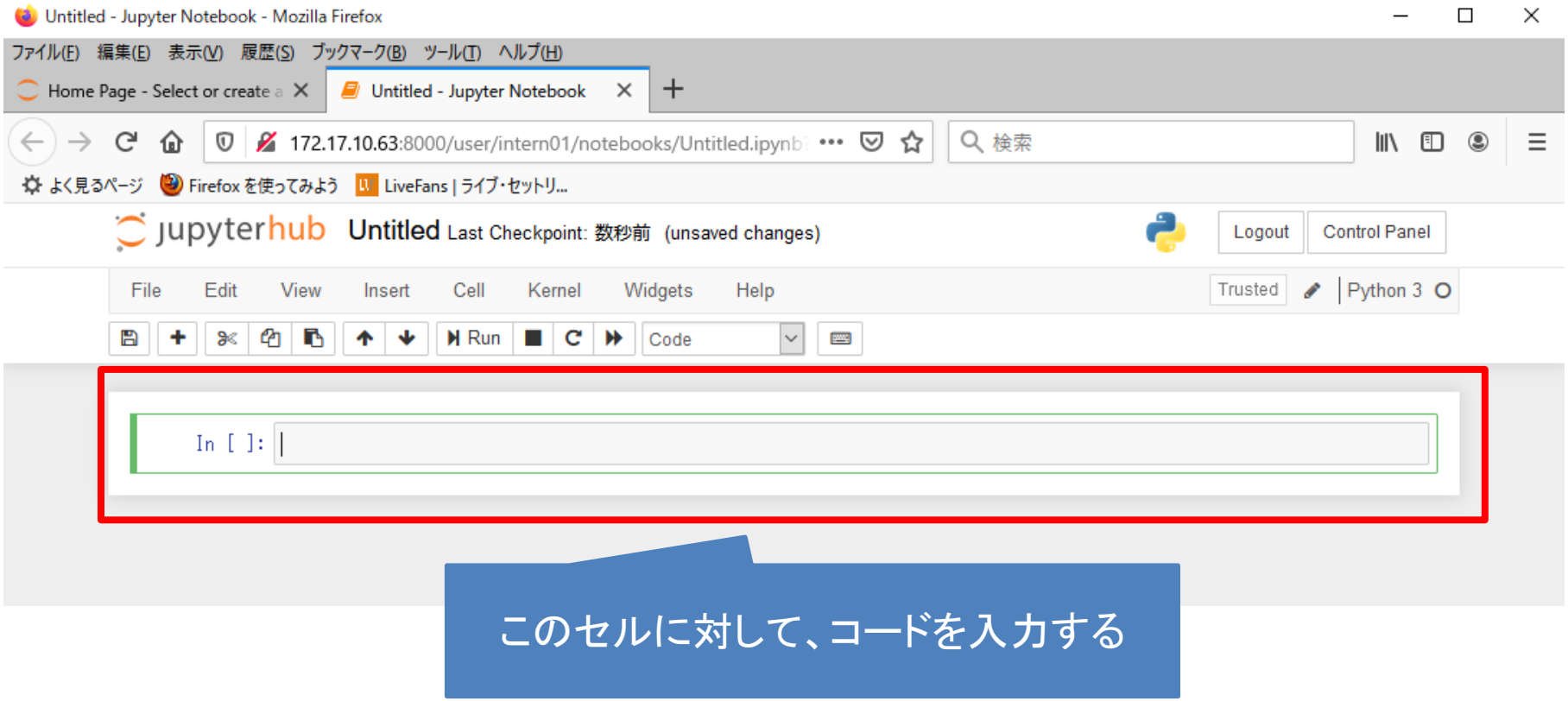
■新規ファイルの作成



“New▼”から”Python3”を選択

Notebookの簡単な操作

■別タブでコード入力を行うための画面が開く



Notebookの簡単な操作

■コードの入力と実行

```
In [ ]: a = 10
        a = a + 1
```

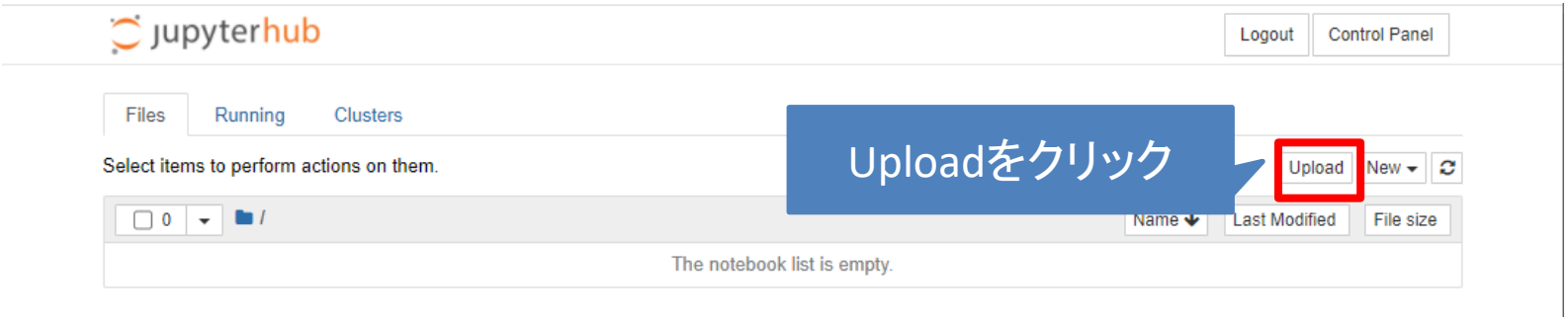
セルにコードを入力し”Shift + Enter”で実行

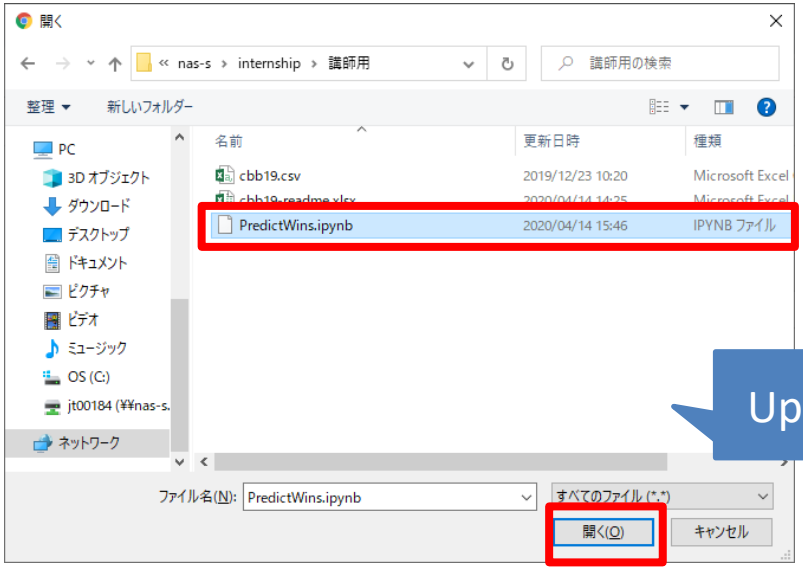
```
In [1]: a = 10
        a = a + 1

In [2]: print(a)
        11

In [ ]:
```

実行結果の表示も可能

Notebookへのファイルアップロード

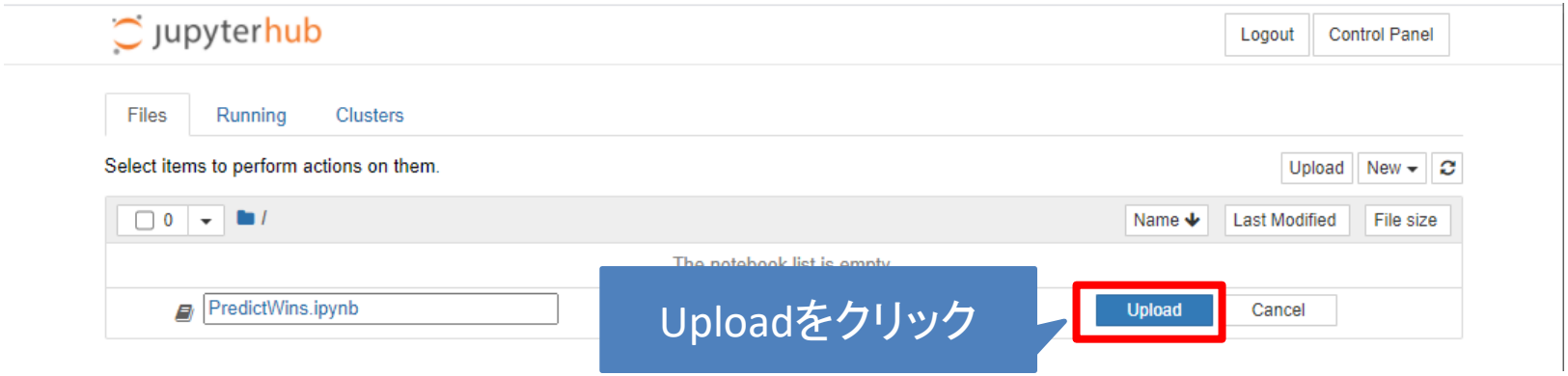
■ファイルのアップロード





Notebookへのファイルアップロード

■ファイルのアップロード



回帰分析の概要
-サンプルプログラムを用いて-

サンプルプログラムの概要

2019年におけるアメリカの大学バスケットボールチームのシーズン中の勝ち数を予測する

■使用するデータセット

– Kaggle(※)で公開されている” **College Basketball Dataset”**から2019年のデータを利用

※Kaggle

Kaggleは企業や研究者がデータを投稿し、世界中の統計家やデータ分析家がその最適モデルを競い合う、予測モデリング及び分析手法関連プラットフォーム及びその運営会社である。
2017年3月8日、GoogleはKaggle社を買収すると発表した。
(Wikipediaより)

使用するファイル

・サンプルプログラム(PredictWins.ipynb)

・使用するデータ(cbb19.csv)

補足)データの説明資料(cbb19-readme.xlsx)

上記の3ファイルをチャットでお送りします。
プログラムとデータをNotebookにuploadしてください。

サンプルプログラムの説明

“PredictWins.ipynb”を実行しながら説明します。

予測コンテスト

予測コンテストの概要

テーマ：東京23区の賃貸物件の家賃を予測する

■使用するデータ
- とあるサイトで公開していた賃貸物件のデータ
  - に少しだけ手を加えたもの

■配布データの一覧
- 分析用データ
  - **rental_housing_train.csv**(30,970件)
- 予測対象データ
  - **rental_housing_test.csv**(500件)
- 参考
  - rental_housing_train_original.csv(分析用データの元データ)

分析用データを用いて、予測モデルを作成する。
その予測モデルを用いて、予測対象データに対して家賃を予測する。

まずやること

・配布されたデータを眺める。

・目的変数(今回は家賃)に影響を及ぼしそうな説明変数を見つける。

・その説明変数を数値化する。
※ある項目を二値化する際、値の種類の数だけ項目が追加される。
値のバリエーションが大きい項目を二値化しようとすると、
とんでもない時間がかかることもあるので注意。

・データの加工はPythonで読み込んでからやってもいいし、事前にExcel上で加工したものをPythonに読み込ませてもいい。
ただし、後者はupload忘れに注意。

サンプルプログラム

・「面積」だけを数値化(単位を除去)し、他の項目を削除したデータを使用。

rental_housing_train_menseki.csv
rental_housing_test_menseki.csv

・このデータを使って予測を行うプログラム

PredictRentalSample.ipynb

・配布するのでNotebookにuploadしてください。
動かしながら説明します。

・プログラムの大枠はこのサンプルプログラムから大きく変更する必要はないはずなので、データの加工に注力ください。

精度判定と予測結果の提出について

■予測精度の判定はRMSEを用いて行います。RMSEの値が小さいほど予測の精度が高いと認識します。

■評価データに対する予測結果を提出していただければ、その結果に対して本人にのみRMSEをお知らせします。

■予測結果を提出する際は、元データに付されているidと予測値が対になったCSV形式で提出ください。

ファイル名は”XXXX_n.csv”としていただけると助かります。

XXXX：お名前　n：提出回数

例）佐藤さんの3回目の予測結果

sato_3.csv

その他あれこれ

■予測に有益であると思えば、どのようなデータを追加していただいても構いません。

– インターネットで公開されているオープンデータ

» データを追加する場合は、分析用データ、予測対象データの両方に追加する必要があります。

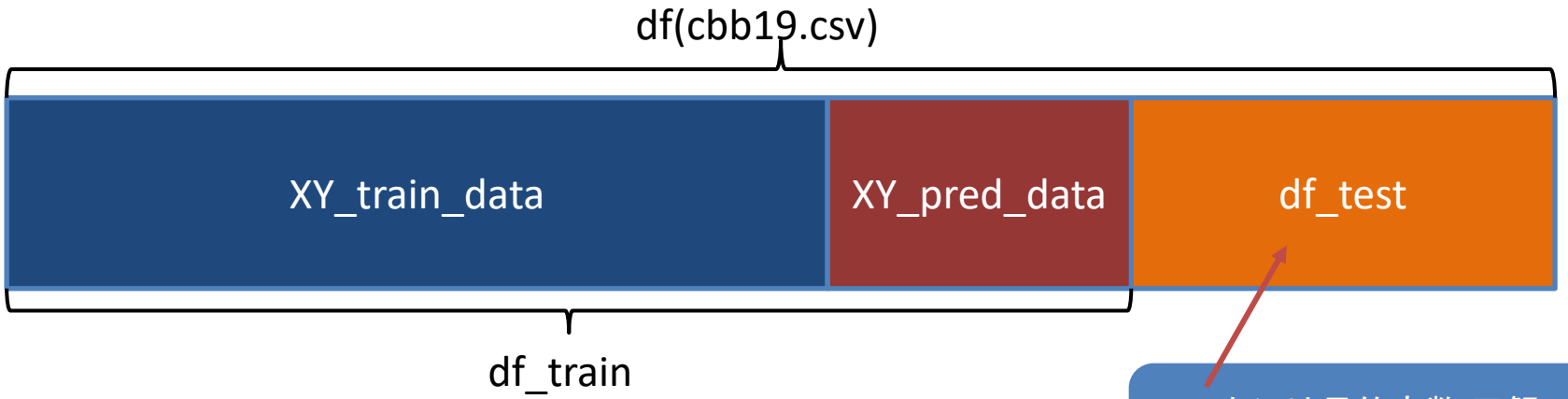
■また、サンプルとは異なるアルゴリズムを使用していただいても構いません。

■精度を上げることは一つの目標ですが、予測結果に納得感が得られるか、にも注意を払ってください

■最後に、工夫した点や苦労した点等を各自発表していただきます。そのための資料準備は不要ですが、長くても5分程度で発表できるようイメージしておいてください。

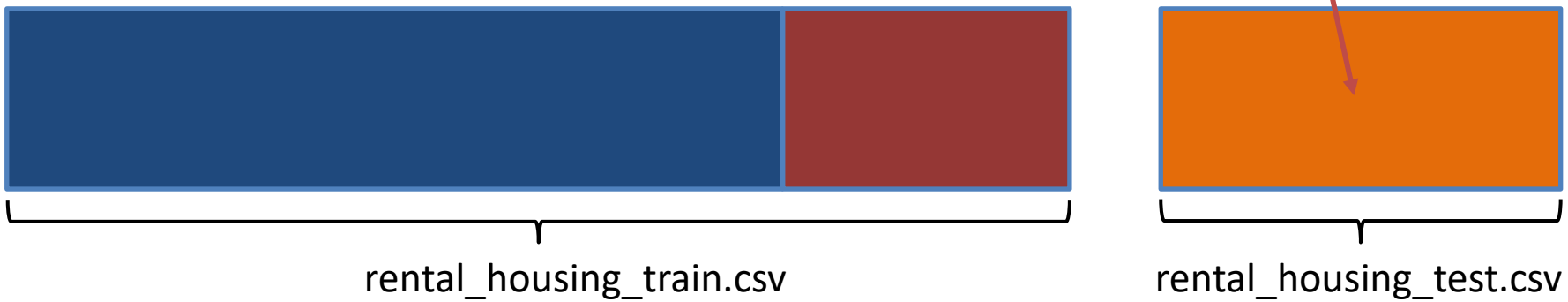
(補足)データの関係

■Cbb19の場合



こっちには目的変数(正解)の値があるが、こっちにはない。

■家賃予測の場合



(補足)データ加工の際の注意点

・データの形式について

trainデータとtestデータは同じ形式でなければいけません。

例えばtrainデータは

Id	賃料	間取り
1	70000	1DK
2	80000	2DK
3	75000	1LDK

となっていたとします。
しかしtestデータにはたまたま1LDKのデータがなかった場合、
“間取り”を2値化(get_dummies())するとtrainデータにのみ
“間取り_1LDK”という列ができてしまいます。

(補足)データ加工の際の注意点

その結果、trainデータとtestデータの形式(列数)が異なる
という結果になり、予測しようとするとエラーが発生して
しまいます。
カテゴリ変数のある値がtrain、testの一方にしか存在しない、
というのは容易に想定できるので、あらかじめ注意を促して
おくべきでした。

対応としては
　①手動で足りない列を全て0で追加する
　②2値化するまえに一度二つのデータを連結して
　　2値化後に改めてtrainとtestに分離する
というような方法があります。