

# Pythonにおける プログラミング経験について

---

名古屋工業大学大学院修士課程2年

福田 太一

# 自己紹介

---

生年月日	1996年8月27日愛知県生まれ
所属	名古屋工業大学大学院・博士課程前期2年 舟橋研究室
趣味	料理, サイクリング, TVゲーム, ストレッチ
メールアドレス	<a href="mailto:f-taitai@i.softbank.jp">f-taitai@i.softbank.jp</a> , guhaabeshi@gmail.com
一言	データサイエンスの知識を深めるべく日々邁進中です. (現在はPythonに力をいれています.)

# 習得したプログラミング言語

言語	経験年数	習熟度 (5段階)	経験の具体例
R	2年間	★ ★ ★	<ul style="list-style-type: none"><li>ロジスティック回帰</li><li>主成分分析</li><li>数理モデルの実装</li><li>時系列分析 (初歩)</li></ul>
Python	8ヶ月	★ ★ ★	<ul style="list-style-type: none"><li>主成分分析</li><li>xgboostを用いたGBDT 回帰モデルの実装など</li></ul>

# これまで作成した Pythonによる成果物

---

1. 求人の応募数を予測するプログラム
2. 食事の評価スコアを分類するプログラム
3. 修士の研究に必要なデータを自動収集するプログラム
4. KaggleのコンペTitanicで使ったプログラム

# 1. 求人の応募数を 予測するプログラム

# プログラムの概要

---

- インターンの選考テストにて作成したプログラム(2020年11月作成)

- ▶ ソースコード

- [Python/di.ipynb at main · bond2580/Python \(github.com\)](https://github.com/bond2580/Python/blob/main/di.ipynb)


- 実務で使用するデータを用いた回帰モデルを作成

# 取り扱ったデータの概要

---

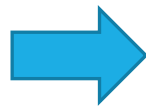
とある求人サイトの各求人先における特徴量  
(勤務地, 給与, 残業の有無, 福利厚生など) と応募者数  
を記録したデータ (15853行 × 265列).

# 文字データの前処理

自分の作成した回帰モデルに文字データは使用できない  
 **ダミー変数に変換する (ダミー化する)** 必要がある。

- ダミー化の例

顧客No.	朝食
1	ご飯
2	パン
3	麺類



顧客No.	朝食: ご飯	朝食: パン	朝食: 麺類
1	1	0	0
2	0	1	0
3	0	0	1



# 数字データの前処理

---

- 欠損値の処理

- 給与, 開始・終了時刻などの特徴量は**応募者数の予測に寄与すると考え**, それぞれ中央値で置換.

- それ以外の欠損値は0で置換.

- 分散が0の特徴量を除去

- 分散が0, すなわち値が全て同じ特徴量は予測に影響しない.

# 使用した回帰モデル

---

## 勾配ブースティング木 (xgboost) [1]

1. 複数の決定木による学習.  $m$ 番目の木では $m-1$ 番目の木で上手に推定できなかった部分に重みをつけて学習.
2. それぞれの決定木の精度の高さに応じて各決定木の予測値に重みを付けて集約.

[1]Chen, Tianqi, and Carlos Guestrin, “Xgboost: A scalable tree boosting system.” Proceedings of the 22nd, acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016.

# 最終的な精度

---

平均二乗平方根差 (RMSE)	0.5899
決定指数 ( $R^2$ )	0.3830

最終的な精度は高いとは言えない数値であった.

他のモデルを試す, パラメータのチューニング範囲の拡張といった  
改善点あり.

## 2. 食事の評価スコアを 予測するプログラム

# プログラムの概要

---

- インターンの選考テストで作成したプログラム. (2021年2月作成)

- ソースコード

- [Python/hac.pdf at main · bond2580/Python \(github.com\)](#)

- 食事に関する特徴量から食事の評価スコア (1～4) を分類する.

- 手法・実験結果の詳細

- [Python/hac.pdf at main · bond2580/Python \(github.com\)](#)

### 3. 研究に必要なデータを 自動収集するプログラム

# プログラムの概要

---

現在研究しているCOVID-19の流行を予測するモデルの  
トレーニングデータ[2]の収集と加工を自動で行うプログラム。  
(2021年4月作成)

## ●ソースコード

[virus/Auto\\_correct.py at master · bond2580/virus \(github.com\)](https://github.com/bond2580/virus/blob/master/virus/Auto_correct.py)

[2][新型コロナウイルス 国内感染の状況 \(toyokeizai.net\)](https://toyokeizai.net)

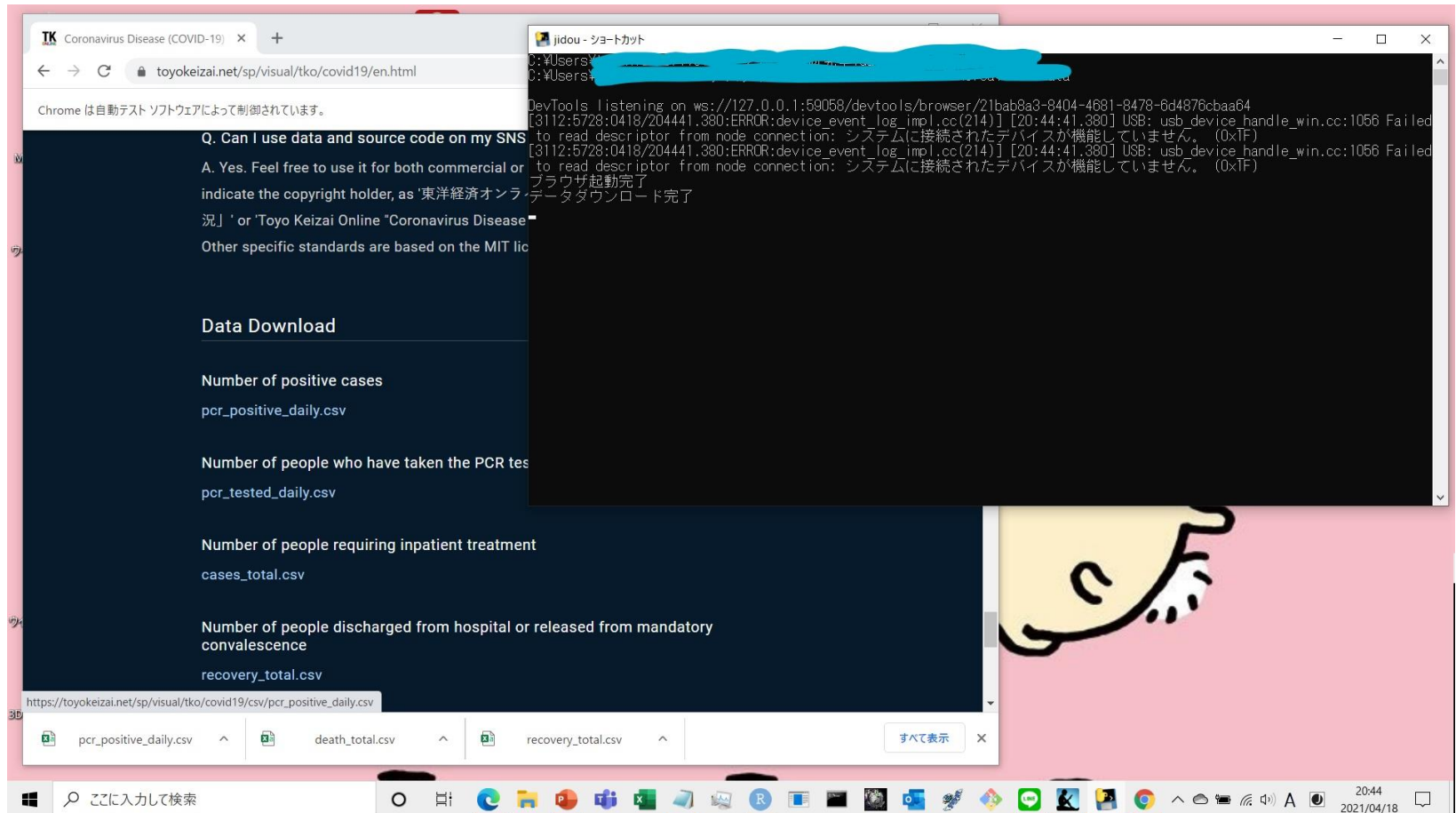
# プログラムの流れ

---

1. Webブラウザを操作するライブラリSeleniumを使用してChromeを起動し, URLを入力
2. ダウンロードするcsvファイルのリンクを自動でクリック
3. ダウンロードしたcsvファイルを読み込んでデータを加工



# プログラム実行中のスクリーンショット (pyファイルからexeファイルを作成して実行)



## 4. Kaggleで使⽤した プログラム

# コンペ: Titanic

---

[Titanic – Machine Learning from Disaster | Kaggle](#)  
で作成したプログラム (2021年4年)

➤ ソースコード

[Python/titanic.py at main · bond2580/Python \(github.com\)](#)

タイタニック号で生き残った乗客とそうでない乗客  
(目的変数:Survived) を分類する.

# データの前処理

---

- **文字データ**

ダミー化したときにサイズが膨大にならない  
Sex, Embarkedのみを残し, ダミー化.

- **数字データ**

欠損値をすべて特徴量ごとの中央値で置換.

# 使用したモデル (詳細は割愛)

---

- ロジスティック回帰モデル
- サポートベクターマシン (SVM)
- 決定木
- ランダムフォレスト

これらのモデルのパラメータを**ランダムサーチ**によってチューニングし, 最も高精度なモデルを決定

# 最終的なKaggleスコア

---

0.7655  
(25200位/32225人)

最も高精度だった モデル	最適化されたパラメータ
	決定木の深さ
決定木	9

もっと高いスコアを目指して頑張りたい。

# コンペ: Tabular Playground Series – May 2021

---

[Tabular Playground Series - May 2021 | Kaggle](#)

で作成したプログラム (2021年4年)

➤ ソースコード

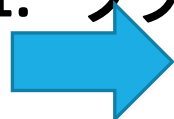
[Python/playground.ipynb at main · bond2580/Python \(github.com\)](#)

50種の特徴量からClass\_0 ~ Class\_3の4つのクラスを分類  
モデルはxgboostを使用


# データの前処理

---

## 1. クラス間のサンプル数の調整

 下調べの結果, 大きな偏りを発見

## 2. 次元削減(線形判別分析)

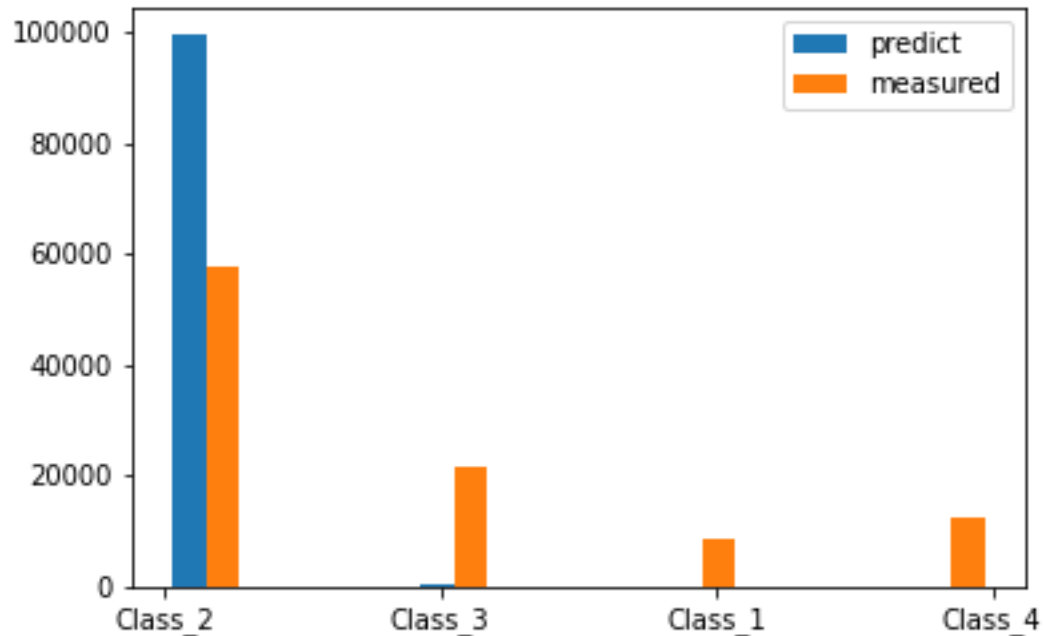
 特徴量50種類はさすがに多いので2種類に削減



# クラス間サンプル数の調整

教師データのクラスが大きく偏る

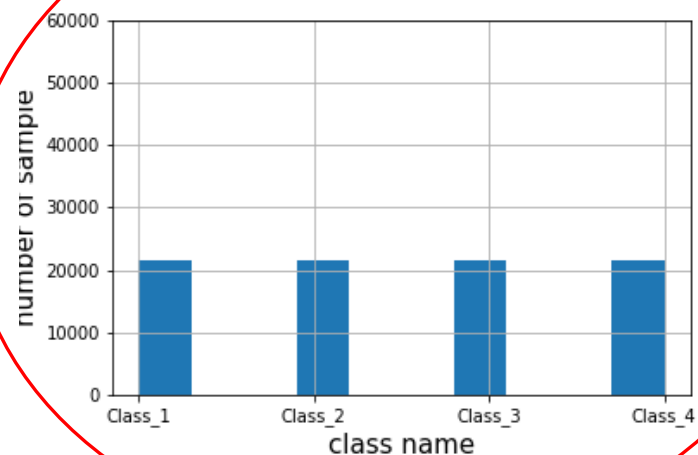
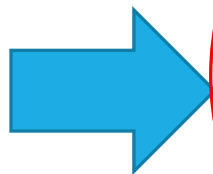
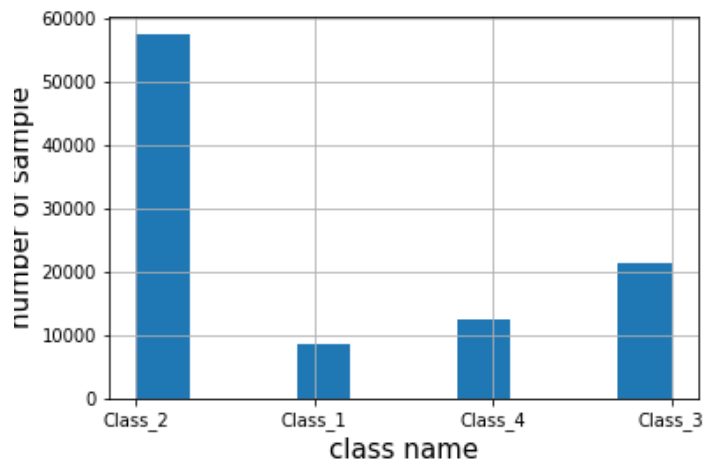
→ 学習後のモデルの予測結果も然り



教師データのほとんどをClass\_2と識別している

# サンプル数調整結果

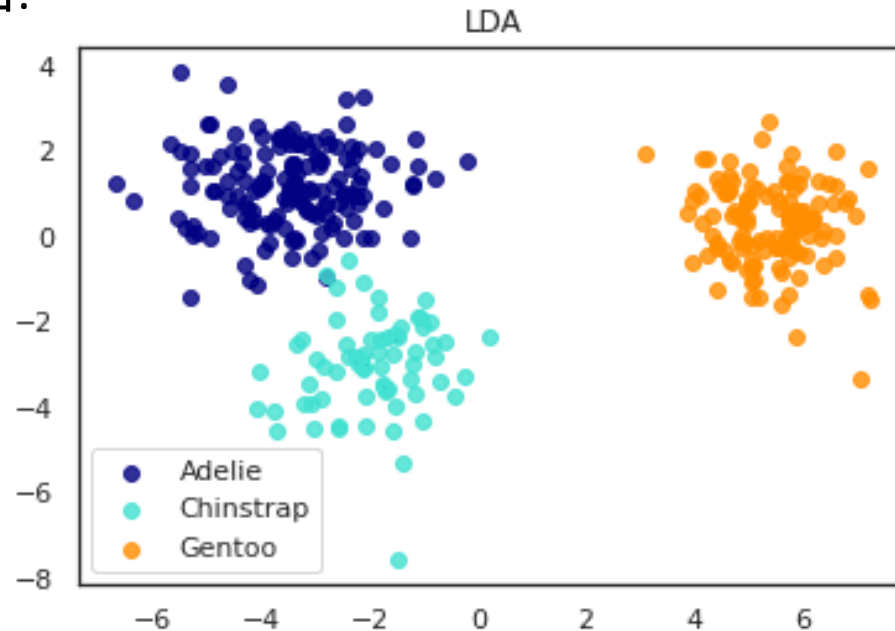
Class\_2のサンプルを半分以下に減らし  
他のサンプルをClass\_2と同じにする



クラスごとのサンプル数の偏りが解消

# 線形判別分析 (LDA)

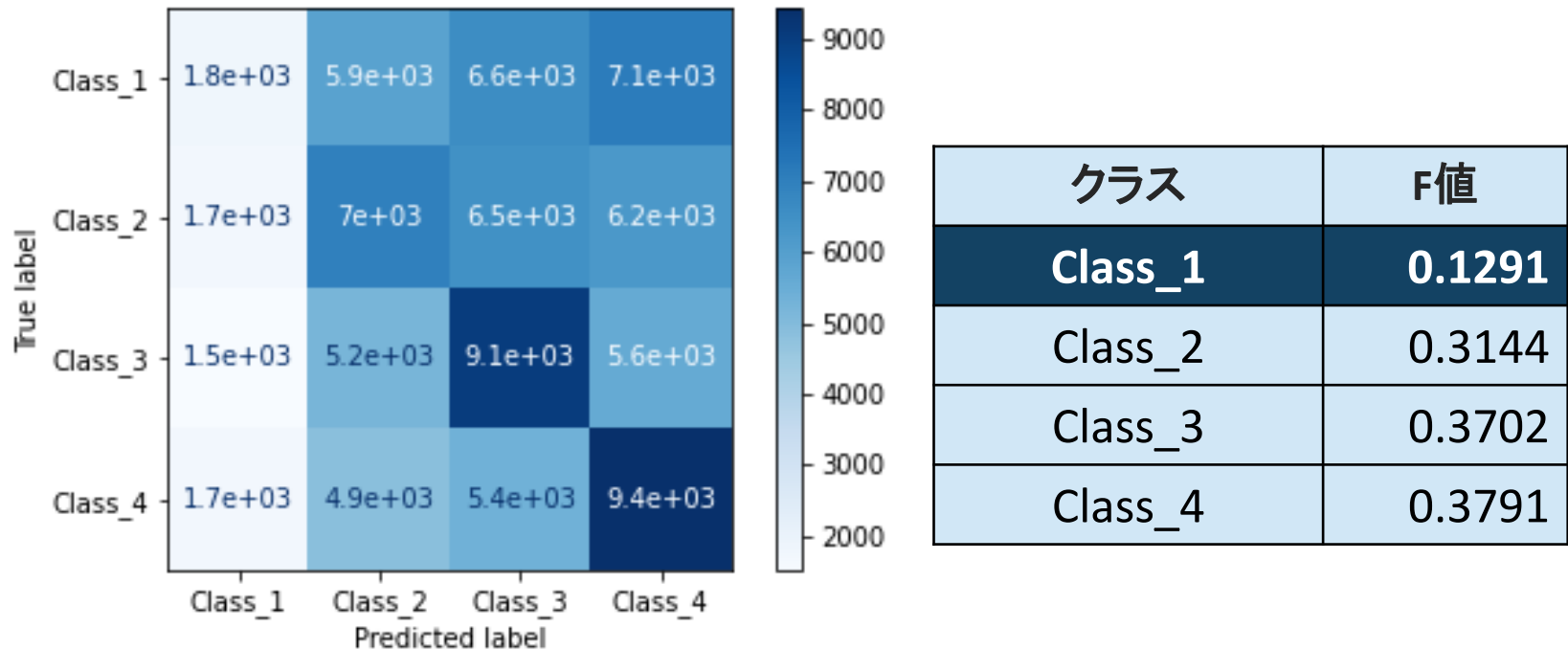
適応前の特徴量よりも次元数が少ないかつクラスを分類しやすい特徴量を抽出.



線形判別適応例

引用元: [線形判別分析\(Linear Discriminant Analysis\) LDA 次元削減 – S-Analysis \(data-analysis-stats.jp\)](https://data-analysis-stats.jp/)

# 学習データの予測結果の ヒートマップとF値



サンプル数調整で変動が最も大きかった  
Class\_1の相対的な識別率が低い

# 最終的なKaggleスコア

---

1.38629

(1051位/1097人)

コンペに投稿されているdiscussionを見て  
高い精度を出す方法を参考にしたい



The background of the slide is a close-up photograph of various autumn leaves. The leaves are in shades of brown, tan, and light green, with some showing signs of decay and discoloration. The lighting is soft, creating a gentle, nostalgic atmosphere. The leaves are scattered across the frame, with some overlapping others, creating a textured, layered effect.

# 最後に

---

**最後までご覧いただきありがとうございました。**