

実世界情報実験1

D1/D2クラス

画像処理テーマ

島田伸敬

12:00より開始

Google Colaboratory (<https://colab.research.google.com/>) を用いて実験を行います。Googleアカウントが必要です。

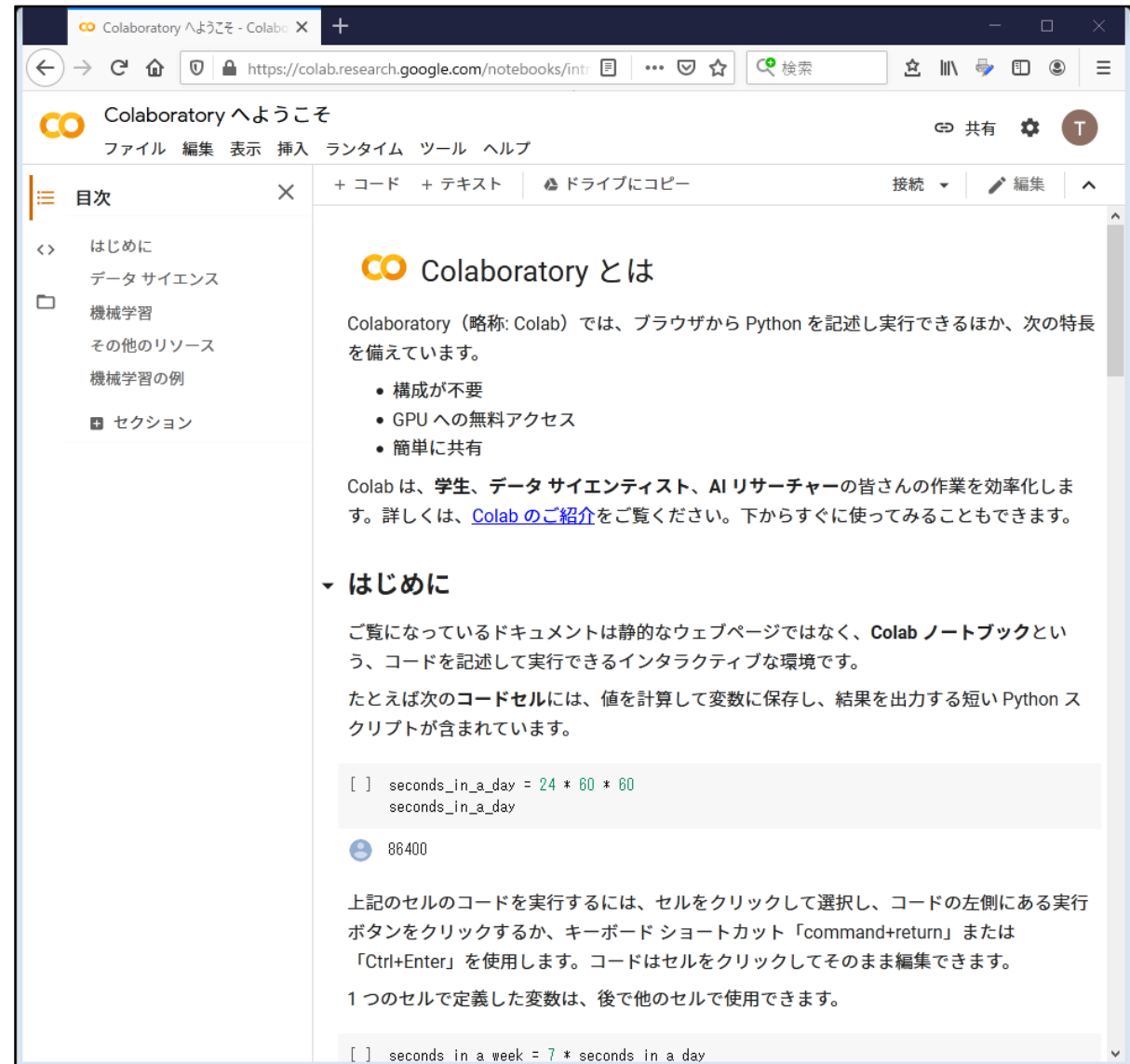
manaba+Rのコンテンツから lesson1.zip をダウンロードしておいてください。

画像処理テーマ

- 画像データの取り扱い方や概念を解説し、簡単な画像処理実験を行う。
- 実行環境としてGoogle Colaboratory (<https://colab.research.google.com/>)上のPythonを使うのでGoogleアカウント(Gmailのアドレス)が必要となる。
- Colaboratory用のノートブックをmanaba+Rからダウンロードし、課題に取り組む。

Google Colaboratory

- URL:
<https://colab.research.google.com/>
- 利用にはGoogleアカウントが必要
- ブラウザのみでPython環境での実験ができる。
- プログラムコードや実行結果を「ノートブック」としてダウンロードできる。



画像処理実験

- Google Colaboratory上でノートブックを開き、課題に取り組む。
- Colaboratory環境には各種ライブラリが整備されており、OpenCVという画像処理ライブラリを利用した画像処理実験を行う。

lesson1.ipynb ☆

ファイル 編集 表示 挿入 ランタイム ツール ヘルプ 最終保存: 18:41

接続 編集

目次

- 2022年度 実世界情報実験 1 画像処理テーマ
 - 課題説明
 - 第1課題ファイル
 - 画像の読み込みと表示
 - 実行後にチェック!
 - 以下実際にコードを書いて実行してみよう
 - 検索してみよう
 - 検索のテクニック
 - カラー画像の濃淡化
 - 課題 1-1
 - 課題 1-2
 - 画像データ
 - 画像データを生成する
 - 課題 1-3
 - チャンネル操作
 - 課題 1-4
 - 課題 1-5
 - OpenCVの画像変換機能
 - 課題 1-6
 - OpenCVの描画機能
 - 課題 1-7
 - ヒント集
 - プログラムが応答しなくなったときの対処
 - カメラデバイスの利用

▼ 画像の読み込みと表示

画像ファイルを読み込んでnotebookに表示する。

以下のリンクをクリックすると画像単体が表示される。これを保存して、それを読み込んで表示してみよう。



OpenCV

PythonでOpenCVライブラリを利用するには cv2 モジュールを import する。また、画像を Google Colab上に表示するには matplotlib.pyplot モジュールを使う。

```
[ ] # 画像処理ライブラリ OpenCV のimport
import cv2

# 画像・グラフ表示ライブラリ matplotlib の import
# ここでは matplotlib内の pyplot モジュールを import し、
# 今後 plt という名前で利用するという設定を行っている。
import matplotlib.pyplot as plt
```

Google Colab上のPythonはクラウドシステム上で動作しているので、そのPythonに画像デー

実験の流れ

1. manaba+Rから課題1のzipファイルlesson1.zipを取得し展開する。
2. Google Colaboratory
<https://colab.research.google.com/> を開く。(zip内にショートカットファイルあり)
3. lesson1.ipynbをアップロードして開く。
4. 画像ファイルをアップロードする。
5. 課題に取り組む（アップロードした画像を利用するプログラムを作成）。
6. lesson1.ipynbや生成した画像ファイルをダウンロードする
7. 次回の授業開始時までmanaba+Rに提出する。

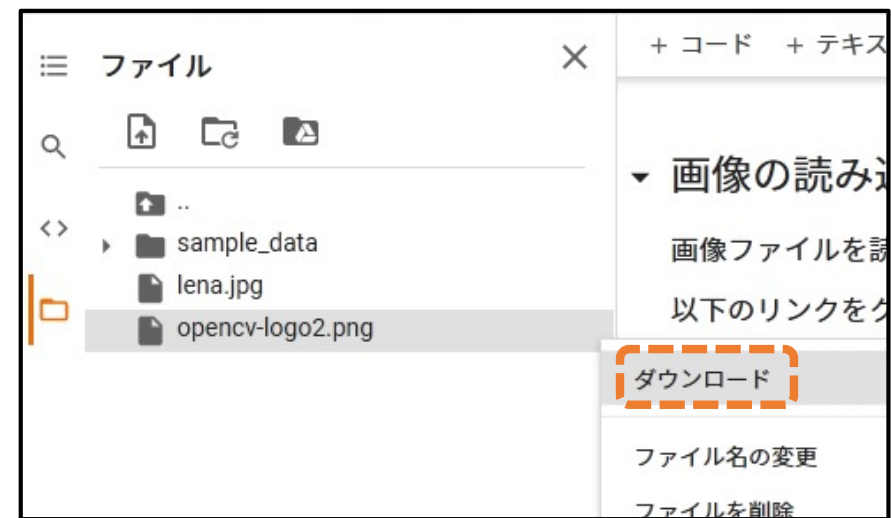
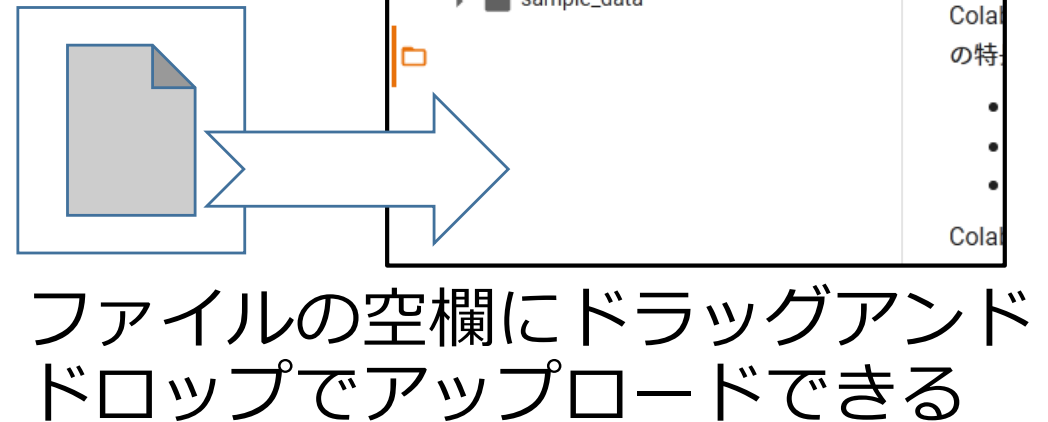
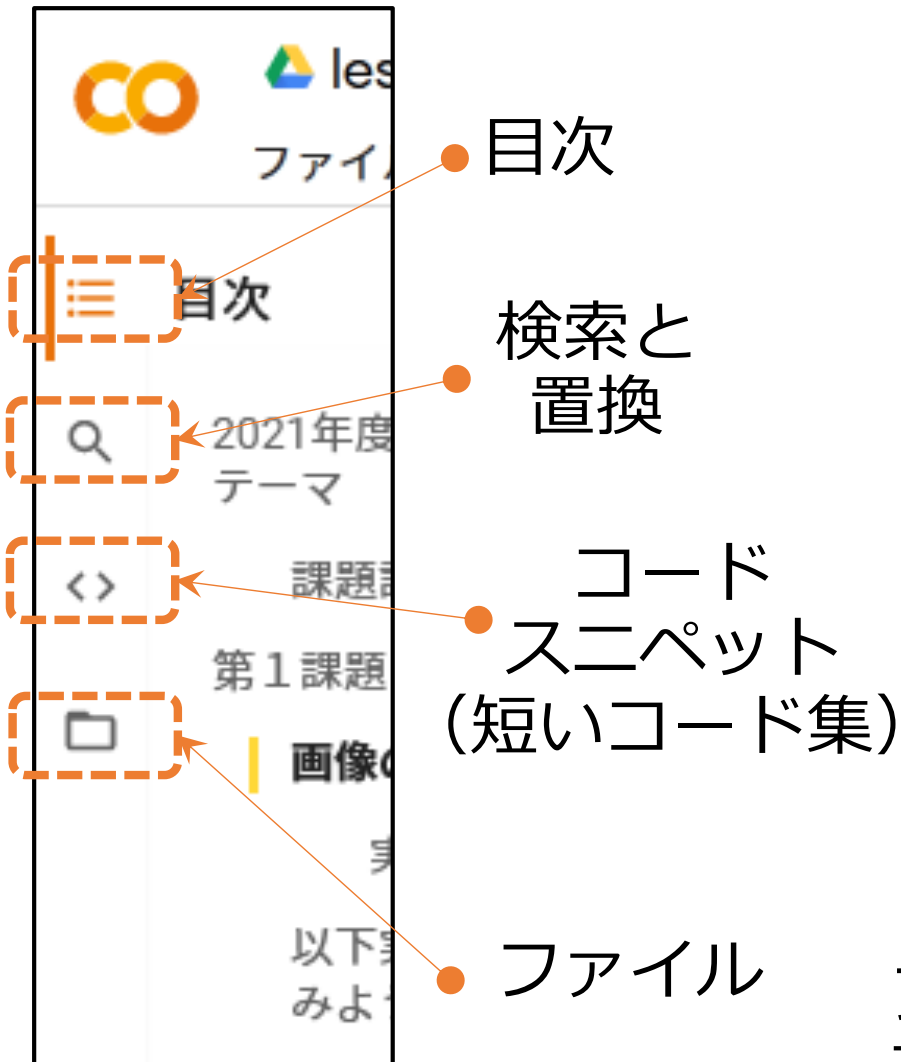
ipynbファイルのアップロード

Google Colaboratoryを開いた画面で「アップロード」タブを選択するか、メニューの「ファイル」から「ノートブックをアップロード」を選択するとアップロード用のフォームが現れる。



フォームにipynbファイルをドラッグアンドドロップすればアップロードできる。

Google Colaboratory 基本操作



ファイル名の右の「…」をクリックすればダウンロードを指示できる

課題提出ルール（現時点）

- 画像処理テーマ第 n 回授業で課題 n の解説を行う。
- 課題 n の一次締切は画像処理テーマ第 $(n+1)$ 回授業日の12:00とする。但し課題6は一次締め切りなし。
- 一次締切までに提出されたものはTAがチェックを行い、結果を受講生本人に公開（フィードバック）する。但し文章で回答する小課題については内容のチェックは行わず、回答があれば10点とする。
 - 小課題それぞれについて10点満点の点数を付ける。これは小課題単体の点数で、その単純和で最終成績を決めるわけではない。例えば課題1-1と課題5-1では難易度が違うので同じ10点でも最終成績に与える影響は異なる可能性がある。
- 課題 n の一次締切内提出物は第 $(n+1)$ 回授業中にTAがチェックする。全提出物について完了したら公開する(各受講生は自分の成績のみ閲覧可能)。
- 全課題の最終締切は画像処理テーマ第7回(最終回)授業の1週間後とする。
- 最終締切までは何度でも再提出できるが、一次締切以後の提出物はフィードバックされないかもしれない。
- TAの負荷、授業の進捗度合いによって上記ルールは変更される可能性がある。

提出についての補足

- ipynbファイル(実行結果も入ったもの)と保存した画像、利用した元画像を提出してもらいます。
- 回答欄のコードセルには `import` や `imread` (必要であれば)等を含め「**そのセルだけを実行すれば意図した状況が作れる**」ようにしておいてください。
(他のセルでimportしたライブラリや他のセルで作成した変数に依存させない)
- プログラムが生成するファイルの名前は課題の指示に従う。
例えば課題1-2なら `lesson1-2.png` とする。追加で保存したい場合は `lesson1-2-1.png`, `lesson1-2-2.png`などのようにする。
- 画像ファイルを保存する課題では、そのファイルを他のプログラムで開いたときにも意図した色で表示されるようにする。(必要であれば`cvtColor()`を使う)