【TechGym】ゼロからはじめる機械学習入門講座「実践ビジネスデータ分析」
サンプルソースの公開場所：https://github.com/techgymjp/techgym_ai/tree/master/　以下

実行環境がない場合は anaconda を install してください。(抜粋版なので問題番号は連番ではないです)

　ビジネスの現場では必ずしも必要なデータがそろっていない場合があり、前処理やクレンジングをしていないデータを扱うことがある。ここでは実際の分析に近い形で処理をすすめて、現場でデータ分析を活用出来るようにしていく。

　データ分析では、例えば顧客から「顧客データを使って売上を上げる施策を考えて欲しい」という依頼が来ることがある。そこで、小売店での営業実績から顧客データは豊富にあると言われたので分析をはじめてみたら、店舗によってフォーマットがバラバラだったり、必要なデータが一元管理されていなかったりする。その場合にはデータを探したりヒアリングしたりすることも求められる。
　今回はあるプログラミング教室の課題を解決する施策についてデータ分析を用いて考えてみる。

　☆設定☆
　あなたはデータサイエンティストとしてある案件を担当することになりました。依頼主はあるプログラミング教室を運営する経営者です。近年の AI ブームによってプログラミングを学習する人も増えて来て、このプログラミング教室は順調に顧客数を伸ばしてきています。しかし、ここ３ヶ月の顧客数が伸び悩んでいます。トレーナーはよく利用する顧客のことは知っているみたいですが、たまにしか来ない人はいつの間にか来なくなってしまうこともあるみたいです。どんな顧客が定着しているのかデータ分析で何か分かったりするのかを調べてみましょう。

【TechGym】ゼロからはじめる機械学習入門講座「実践ビジネスデータ分析」
■4-1：ビジネスデータ分析：cZN9.py
　プログラミング教室の顧客データを扱って分析を行う
□扱うデータを読み込んでどんなデータであるかを確認する
　・プログラミング教室の利用履歴データ(log.csv)
　・会員データ (customer.csv)
　・会員区分データ(class.csv)
　・キャンペーン区分データ(campaign.csv)
　※これらのデータは Github に登録してあるので各自ダウンロードして使用すること
□データを読み込んで気づいたことを考察してみる
■解答は 9DGy.py

■4-2：ビジネスデータ分析：3YnY.py
　顧客データの整形をしてみる
□データが別れていると扱いづらいので、会員データに会員区分とキャンペーン区分を結合する
　(顧客データが主になるので左外部結合になる)
□データの結合前後でデータ件数に変化がないことを確認する
□結合後のデータの欠損値を確認する(うまく結合できないときは欠損値が増えていることがある)
【ヒント】データ結合(左外部結合)：pd.merge(,how='left')
■解答は 0GOe.py

■4-3：ビジネスデータ分析：0YcV.py
　顧客データの集計をしてみる
□会員区分、キャンペーン区分、性別、既に退会済みかどうかについて全体の数を集計する
□入会人数を集計する(入会人数は start_date 列が 2018/4/1~2019/3/31 までの会員である)
　start_data を datatime 型に変換して customer_start というデータフレームを作った後、
　loc[“入会日” > “2018/4/1”]として集計する
□集計した数から分かることを考察してみる
【ヒント】
□集計：groupby(“集計対象”).count()[“会員データ”]
□datatime 型：pd.to_datatime(対象のデータフレーム)
■解答は UC2n.py

■4-4：ビジネスデータ分析：p4OA.py
　顧客の全体像を把握するために、既に退会している会員の情報を除いて、最新月の顧客データを集計してみる
□最新月に在籍していた会員のみのデータフレームをつくって集計する
　end_data を datatime 型に変換して退会日と 2019/3/31 を比較する
　さらに、在籍している会員は end_data が欠損値であるかで判定する
□会員区分、キャンペーン区分、性別について最新月の顧客データの数を集計する
□集計した数から分かることを考察してみる
□欠損値かどうか：df.isna()
■解答は Z3uG.py

■4-5：ビジネスデータ分析：3LpA.py
　利用履歴データでは時間的な要素の分析をすることが出来る。
　例えば、月内の利用回数がどのように変化しているか、定期的に来て活用している会員かなどがある。
　ここでは以下の処理をして顧客ごとの月利用回数を集計する。
□usedate 列を datetime 型に変換する
□年月のみの列”年月”を新しく追加する：df.strtime(“%Y%m”)
□年月と顧客 ID で集計して新しいデータフレーム(利用回数)をつくる
　出来たデータフレームの cloumns を”log_id”と”count”にする
□不要な”usedate”列は削除する：del
□出来たデータフレームの先頭を表示する

【TechGym】ゼロからはじめる機械学習入門講座「実践ビジネスデータ分析」
□顧客毎の利用回数に対して、平均値、中央値、最大値、最小値を集計する
□reset_index(drop=False)を実行して index をつけ直す
□集計した数から分かることを考察してみる
【ヒント】顧客毎の集計：df.groupby(“顧客 ID”).agg(["mean", "median", "max", "min" ])["count"]
■解答は No6O.py

■4-6：ビジネスデータ分析：O1Vv.py
　プログラミング教室の場合、習慣化が継続の重要な要素になると考えられます。利用履歴データから定期的に使用しているユーザーを特定してみる。ここでは毎週同じ曜日に来ているかどうかを調べてみる。
□利用日の曜日を数字に変換する(Monday = 0、…、Sunday = 6 の曜日)：dt.weekday
□ユーザー毎に月別と曜日別で集計を行う(log_id を数える)
　出来たデータフレーム(log_df_weekday)の cloumns を”log_id”と”count”にして先頭を表示する
□顧客毎の各月の最大値を取得して新しくデータフレーム(log_df_weekday)をつくる
□”routine_flag”という列を新しくつくり「0」で初期化する
□各月で同じ曜日に 4 回以上利用しているユーザーに”routine_flag”を 1 としてフラグを立てる
　uselog_weekday の先頭を表示する
□つくったデータフレーム log_df_customer と log_df_weekday を customer_join に結合する
　結合に使うキーは” customer_id”でレフトジョインになる
【ヒント】
□ユーザー毎に月別と曜日別：["customer_id","年月","weekday"]
□4 回以上のときフラグを 1 にする：where(log_df_weekday["count"]<4, 1)
■解答は N3Uo.py

■4-7：ビジネスデータ分析：mB7D.py
　次に会員期間を計算して追加する
□start_date と end_date の差を使って会員期間を計算する
　2019 年 3 月までに退会していないユーザーに関しては、end_date に欠損値が入っている
　欠損値には 2019 年 4 月 30 日として”calc_date”という列をつくる
□”membership_period”とう列をつくり初期化する
□start_data と calc_date の差分を計算して、月単位の時間に変換する
□["mean", "median", "max", "min"]の統計要約量を表示する
□定期利用フラグを集計する
□会員期間のヒストグラムを描画する
□退会ユーザーと継続ユーザーの統計要約量の違いを比較して考察してみる
(ここまでで使用したデータフレームを CSV で保存しておく)

【ヒント】
□start_data と calc_date の差分：from dateutil.relativedelta import relativedelta
□月単位の時間に変換する：年*12 + 月
□退会ユーザー：customer_join["is_deleted"]==1
□継続ユーザー：customer_join["is_deleted"]==0
■解答は 0tNR.py

■4-8：ビジネスデータ分析：zRE5.py
　顧客の利用方法の傾向をさらに調べてみよう。利用履歴を使ってクラスタリングをしてみる
□前回に作成した” customer_join.csv”を読み込んで customer というデータフレームをつくる
□必要な変数["mean", "median","max", "min", "membership_period"]のデータフレームをつくる
□各変数の大きさが違うので、標準化をする
□クラスタ数を 4 として k-means でクラスタリングをして顧客にラベルをつける
□各クラスタに分類されたデータ件数を表示する、またクラスタ毎の平均値を集計してみる
　(得られた結果を考察してみよう)
□クラスタリングに使用した変数は 5 個なので、主成分分析で次元圧縮をして 2 個にして描画する
■解答は I1hD.py

【TechGym】ゼロからはじめる機械学習入門講座「実践ビジネスデータ分析」

■4-9：ビジネスデータ分析：NjE2.py
　クラスタリングした結果をさらに詳しく調べてみる
□クラスタリングでグループに分けた顧客のうち継続ユーザーと退会ユーザーの集計をする
　(結果からわかることを考察してみる)
□同様にクラスタごとに継続利用フラグの集計をする(結果からわかることを考察してみる)

　過去の行動データから翌月の利用回数を予測することを考える
　(ここでは、過去の 6 ヶ月の利用データを用いて、翌日の利用データを予測する)
□利用履歴データ(log.csv)を用いて毎月、顧客毎に集計をして uselog_months をつくる
□当月と過去 6 ヶ月の利用回数をカウントしてデータフレームを表示する
　・今回対象となる年月データをリストにいれる
　・予測データのデータフレームを生成する
　・ループ変数 i の範囲を range(6,11)として for 文で 2018 年 10 月から 2019 年 3 月までを処理する
　・loc[uselog_months["年月"]==year_months[i]]とすると当月の利用回数が分かるので
　　index を count_perd という名前にリネームする
　・ループ変数 j の範囲を range(1,7)として 6 ヶ月分の利用回数のデータを並べていく
　・loc[uselog_months["年月"]==year_months[i-j]とすることで過去 6 ヶ月分の利用回数となる
　・利用回数回数の index を過去にさかのぼっていくごとに count_0,count_1 として結合して
　　データを並べていく(キーは”cunstomer_id”とする)
　・出来たデータフレームを縦に連結していく
□特徴量として会員期間”priod”をデータに付与してデータフレームを表示する
　(会員期間の計算は前問で行ったのと同様にする)

【ヒント】
□データ結合：pd.merge()
□データフレームを縦に連結する：pd.concat
■解答は NR7c.py

■4-10：ビジネスデータ分析：D6Ko.py
　利用回数の予測モデルをつくって利用回数を予測してみる
□新規に入会したユーザーに対してモデルをつくるので、2018 年 4 月以降に新規に入ったユーザーを対象とする
□データを学習データと評価データに分割して、線形回帰モデルをつくる
□学習させて正解率を表示する
□回帰係数を表示してモデルに寄与している変数について考察してみる
□新しいユーザーの 7 ヶ月分の利用履歴を与えたときに、来月の利用回数を予測してみる

【ヒント】
□2018 年 4 月以降：predict_data["start_date"]>=pd.to_datetime("20180401")
□データ分割：sklearn.model_selection.train_test_split(X,y)
□線形回帰モデル：linear_model.LinearRegression()
□回帰係数：model.coef_
■解答は iD8Y.py