moemoe 2016/1/24

＜用意必要なファイル＞

pythonコード：

- toyama.py（column6を担当）
- hakone.py（column4, 5, 8を担当）
- kanazawa.py（column7を担当）
- foreigner.py（column15~28を担当）
- others.py（その以外のcolumnを担当）
- kawase_test.py（為替データの前処理：休日の部分を補完）
- kawase_train.py（為替データの前処理：休日の部分を補完）

その他：

- ホームページからダウンロードした全てのデータおよびsample_submit.csv

＜再現する手順＞

0. 上記のソースコードおよびデータを同じフォルダーにおく
1. sample_submit.csvのファイル名をresult.csvに変更
2. result.csvを二回コピーして、それぞれのファイル名をhakone.csvとtoyama.csvに変更
3. hakone.csvとtoyama.csvの中、日付以外の列を削除して、保存

（ここまでは渡した状態）

4. hawase_test.pyとkawase_train.pyを実行（順番は構わない、cmdでpython XXX.pyを入力すればいい）
5. 残る五つのpythonファイルを実行（順番は構わない、cmdでpython XXX.pyを入力すればいい）
6. 結果はresult.csvの中にある（コンペの最後提出したファイルとほぼ一致*）

*一致しない部分について、column6（富山）はわずかの差があると思う。なぜというと、この前ダウンロードした訓練データを少しいじった（富山の最後一週間のデータを除外するため、極端の値を与え、異常値処理の時捨てさせられた）

*手動的にtarget/target_train.csvの中にある**'16201_若年層', '16201_中年層', '16201_老年層'**三列の、**2015-5-25から2015-5-31まで**の部分を極値（例えば30000）を与えると、元のデータを再現できるはず

＜構築環境＞

- OS X 10.9.5
- python 2.7.9
  - scikit-learn 0.16.1
  - pandas 0.17.0
  - numpy 1.9.2

＜モデリング＞

大きく二種類のモデルを使用した

column7（金沢）、15~28（外国人）：
・一定モデル + 祝日（イベント）効果

その以外の列：
・RidgeCV（Ridge回帰、自動的に一番適切なパラメータを探索するアルゴリズム）
・説明変数
  ・部門2のレポートの中にあるベースモデルの部分をご参照ください
  ・箱根&富山については調整があるので、それぞれのスライドをご参照ください
・前処理について、提出したスライドをご参照ください

＜備考＞

・乱数は使っていない
・外部データはほぼ使っていない（祝日&イベント日の情報のみ）