hait_analysis_exercise について

挑戦課題

https://deepanalytics.jp/compe/27?tab=compedetail に挑戦しました。

手法

定性的な変数が多かったので、データの前処理にほとんど時間を費やしました。

特徴量が非常に多いので PCA によって次元削減をし、Lasso によって線形回帰をしました。

結果

RMSE が 3700 台となり、応募件数 3154 件中 160 位となりました。

ソースコード解説（上から）

In[2]

データの一部が分割されて配布されていたので、一つにまとめて新しいファイルに書き込みました。pd.DataFrame 上で結合しても良かったのですが、再利用が高いと思いファイル作成にしました。

Out[3]

既に特徴量が多めで、欠損値等はないことがわかりました。

In[6],In[7],In[8]

ここで計算できるように変数を全て数値化しています。One-hot-encoding をした変数は次元が大きくなりすぎたので PCA をかけることにしました。サラッと書いていますが出場選手や放送局のダミー変数化は少々大変でした。

In[10]

df についてですが,2014 年のデータについて予測するので、'year'が教師データにあってもしょうがないので捨てました。また df の対戦カード（チーム名）と df_cond の出場選手は多重共線性があり

そうなので、技術的に面白そうな出場選手の方を採用しました。ただしこれは次元が大きくなりすぎるので、素直にチーム名を説明変数にしたほうが、モデルの精度は良かったかもしれません。df_cond についてですが、試合が終わるまでその数値がわからない試合スコアは捨てました。これら以外の変数は全て数値化しました。

In[13]

ここからデータ分析です。本当はニューラルネットワークでもモデルを作って、その平均を取るアンサンブル学習でもしたかったのですが私のパソコンは一昔前のものなのでスペック的に Lasso のみで予測することにしました。セルには現れていないですが、GridSearch を何回か行って最適パラメータが候補パラメータ集合の境界値にならないようにしました。

In[15]

予測値を、その試合が行われているスタディアムのキャパで上から抑えました。

考察

Pandas は get_dummies でダミー変数を自動で作成できるものの、今回の出場選手のような場合にはある程度工夫しないとだめでした。私が知らない pandas の機能を使えば、もっと楽に書けることがあるようなところも泥臭くコードを書いてしまった部分もあったのかもしれません。もっと上位に行くためには線形回帰では限界があるのかなと思って SVR(kernel=rbf)でもやってみたのですが、今回はこちらのほうがいいスコアを出しました。もう少しまともなパソコンが手に入ったらディープラーニングでも予測モデルを構築してみたいです。