1st Place Solution

~ SIGNATE Career Up Challenge ~

Agenda

1. 自己紹介
2. コンペティション概要
3. 解法紹介
    a. 学習・予測フロー
    b. 評価関数
    c. post-processing
4. コンペ振り返り
5. まとめ

Agenda

Agenda

1. 自己紹介
2. コンペティション概要
3. 解法紹介
    a. 学習・予測フロー
    b. 評価関数
    c. post-processing
4. コンペ振り返り
5. まとめ

Agenda

2.コンペティション概要

中古自動車の価格予測

中古自動車の情報をもとに中古車価格を予測する回帰タスク。評価指標がMAPEであるため、厳密には価格の安いものを大きく外さない予測アルゴリズムを考える必要がある。

データ概要
(一部省略)

目的変数

price = 中古車価格($)

カテゴリ変数

column	unique	isnull
region	372	-
state	51	○
condition	6	-
type	13	○
fuel	5	○
manufact	125	○

量的変数

odometer : 走行距離

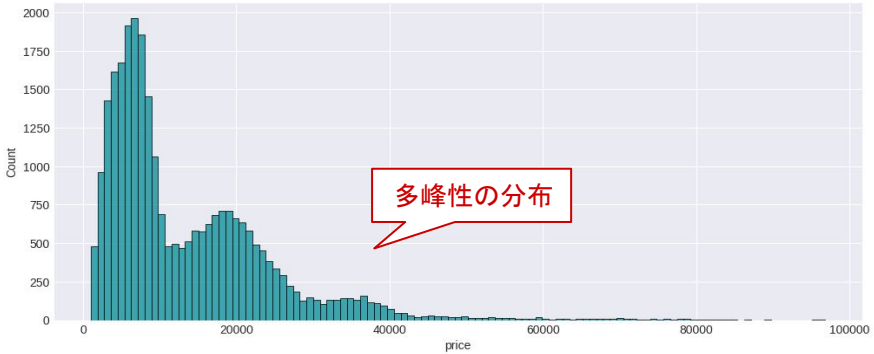
year : 販売年

cylinders : 気筒数

評価指標

$$\mathrm{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| (\%)$$

中古車価格の分布



2.コンペティション概要

コンペ序盤に考えたこと

ベースラインはGBDT系のシンプルなもの、カテゴリ変数はTarget encodingで処理。

① 評価指標（MAPE）について

1. 価格の低いレコードの予測を大外ししないのが重要
2. lossは下に外すことに寛容なため、予測値は実測値に比べて左に寄る

② 特徴量作成

1. 表記ゆれや外れ値などの処理が必要
2. 水準の多いカテゴリ変数は Target-encodingによる処理が効率的
3. ドメインに基づく急所をさせるような特徴量作成は自分には無理

③ バリデーションとモデルの選択

1. 車種(type)により分布が異なるため foldごとのtype割合を均一にする
2. MAPE+GBDT系だと価格の高いレコードの予測は難しい
3. NN系を採用するならカテゴリ変数を Embeddingで処理したい

Agenda

1. 自己紹介
2. コンペティション概要
3. 解法紹介
    a. 学習・予測フロー
    b. 評価関数
    c. post-processing
4. コンペ振り返り
5. まとめ

Agenda

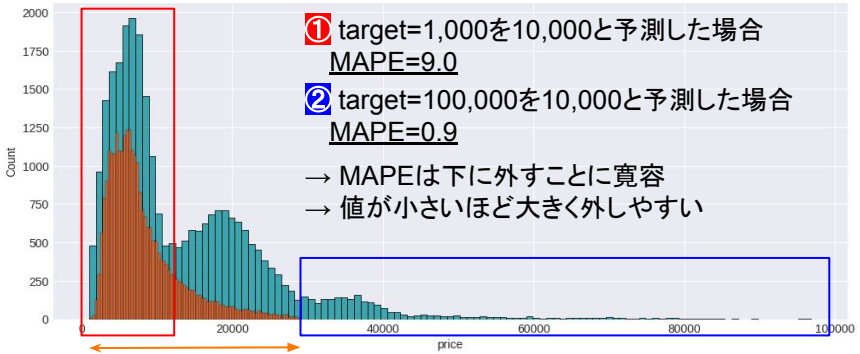
3.解法紹介
MAPEについて

MAPEをlossとしてLightGBMを学習させると、GOSS(Gradient-based One-side Sampling)のアルゴリズムから『残差が大きい(=サンプリング主対象) ≒ targetの値が小さいレコード』という関係になりやすい。

Algorithm 2: Gradient-based One-Side Sampling

Input: I: training data, d: iterations
Input: a: sampling ratio of large gradient data
Input: b: sampling ratio of small gradient data
Input: $loss$: loss function, L: weak learner
models ← {}, fact ← $\frac{1-a}{b}$
topN ← a × len(I) , randN ← b × len(I)
for $i = 1$ ***to*** d **do**
  preds ← models.predict(I)
  g ← $loss$(I, preds), w ← {1,1,...}
  sorted ← GetSortedIndices(abs(g))
  topSet ← sorted[1:topN]
  randSet ← RandomPick(sorted[topN:len(I)], randN)
  usedSet ← topSet + randSet
  w[randSet] × = fact ▷ Assign weight $fact$ to the small gradient data.
  newModel ← L(I[usedSet], − g[usedSet], w[usedSet])
  models.append(newModel)

中古車価格の分布



❶ target=1,000を10,000と予測した場合
MAPE=9.0

❷ target=100,000を10,000と予測した場合
MAPE=0.9

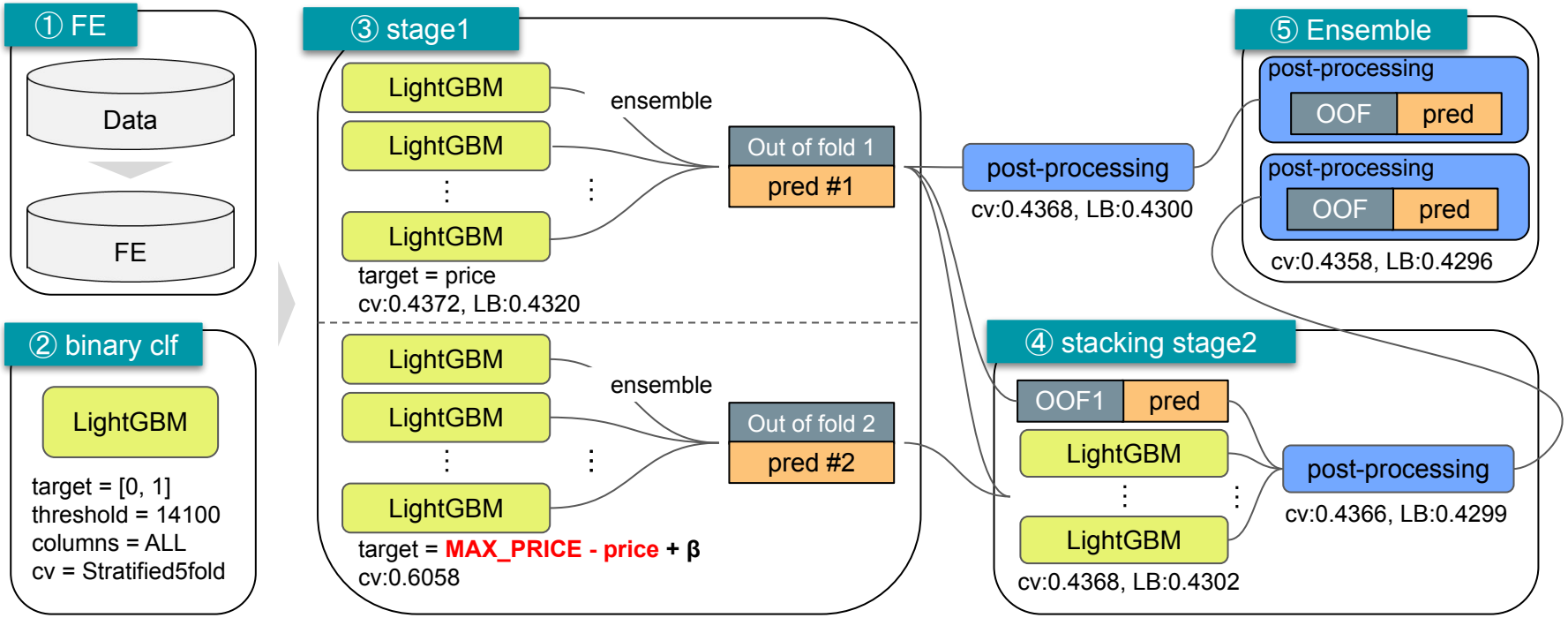
→ MAPEは下に外すことに寛容
→ 値が小さいほど大きく外しやすい

⇒ 本コンペの戦略

1. targetを適切に変換し、値が大きいレコードのサンプリング頻度が高いモデルを作る。
2. 値の大きいレコードについて、後処理のリスクが低い。(分類タスクでいうCalibrationをイメージ)

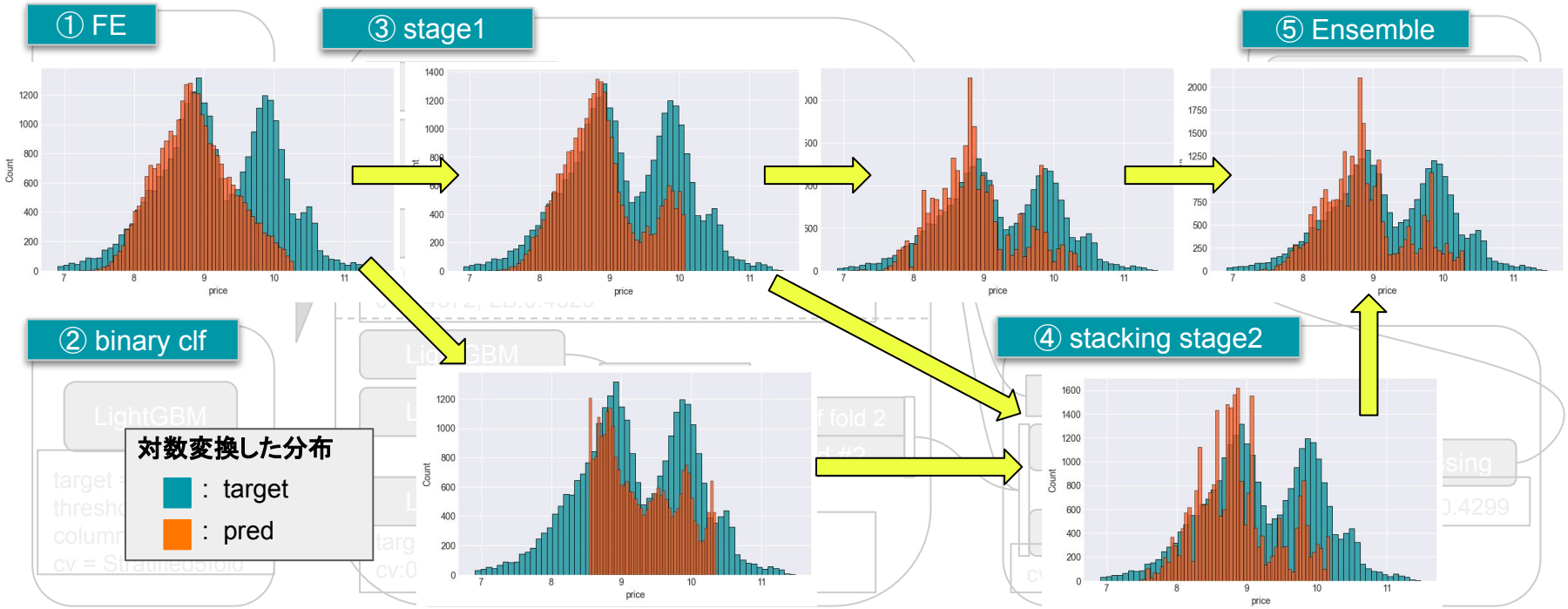
3.解法紹介
学習・予測の流れ

2層のStackingを採用しモデルはLightGBMのみを使用。LightGBMの予測値をAffine変換しEnsembleすることで精度を保持しつつ分布の裾を広げた。



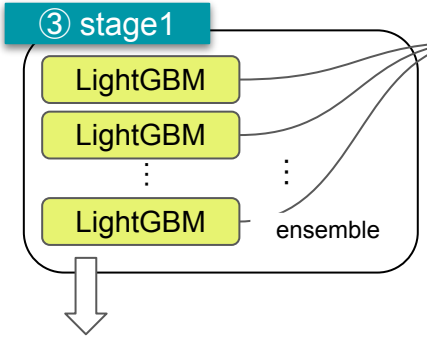
3.解法紹介
予測値の分布の変化

stackingとtargetの変換は分布の範囲を広げアンサンブルに多様性を生んでいる。またbinary taskの確率値を特徴量に加えることで複数の"峰"を予測できている。

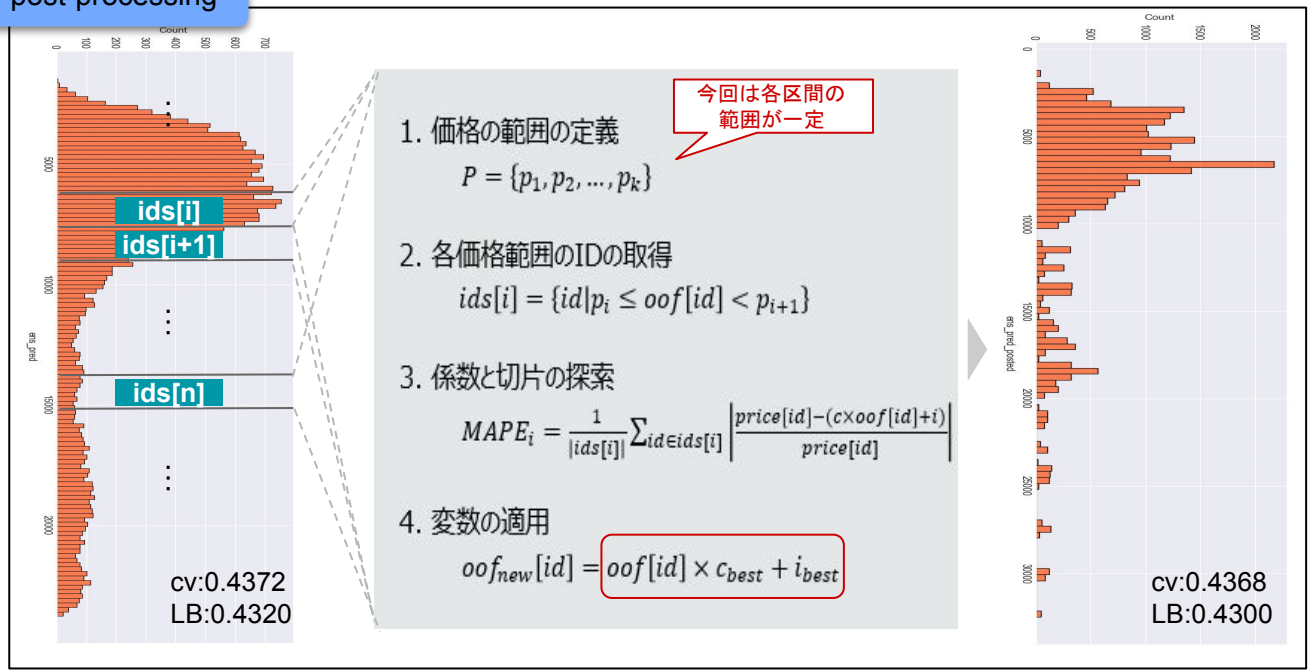


3.解法紹介
post-processing

モデルの出力を一定の範囲で区切り、各区分ごとにAffine変換を行った。coeff(線形変換)とintercept(並行移動)を探索し、out of foldに対して最も性能がよかったものを採用した。



post-processing



【stage1段階の課題】

- ❑ 予測値の最大値が低い
- ❑ アンサンブルを想定した時モデルの多様性に欠ける
- ❑ target > 15000のレコードのRMSEは非常に大きい
- ❑ MAPEの誤差が非常に大きいのはtarget < 1000のレコード

Agenda

1. 自己紹介
2. コンペティション概要
3. 解法紹介
    a. 学習・予測フロー
    b. 評価関数
    c. post-processing
4. コンペ振り返り
5. まとめ

Agenda

振り返り

タスクは非常にシンプルでかつ特徴量作成も複雑になりずらいものだったためLBスコアはモデル選択や学習全体の設計によるものだと感じた。

■ 個人的に効いた仮説

- カテゴリ変数はGBDTの学習効率を上げるためにtarget-encodingによる処理が最適ではないか。
- 多峰性のある分布(今回は3峰)の予測に対して評価指標がMAPEの時、GBDT系単一のモデルでは最小の峰に予測値が寄ってしまうだろう。
- pipeline全体を通して学習データを満遍なく活用するには、適切なloss関数の採用と適切なtargetの変換が必要だろう。
- GBDT系は層を重ねた方が予測範囲が広がるだろう。

■ 改善点

- 予測値の分布形状からスコア向上の道筋を立てたが、すべてのコンペで有効な方策ではない。
- 予測分布やデータのサンプリング頻度は複数アルゴリズムのモデルで多様性をだすべき。
- post-processingの変換法では汎化性能が担保されていない。十分なデータ量を持つtrainのパラメータ探索時は検証プロセスも入れるべき。
- 特徴量作成に全く時間をかけられなかった点。

Agenda

1. 自己紹介
2. コンペティション概要
3. 解法紹介
    a. 学習・予測フロー
    b. 評価関数
    c. post-processing
4. コンペ振り返り
5. まとめ

Agenda

5.closing
まとめ

■ 最終順位：1st Place

cv : mape = 0.4358
Public LB : mape = 0.4296
Private LB : mape = 0.4284 (最終subには選べなかったが最高はmape = 0.4281)