



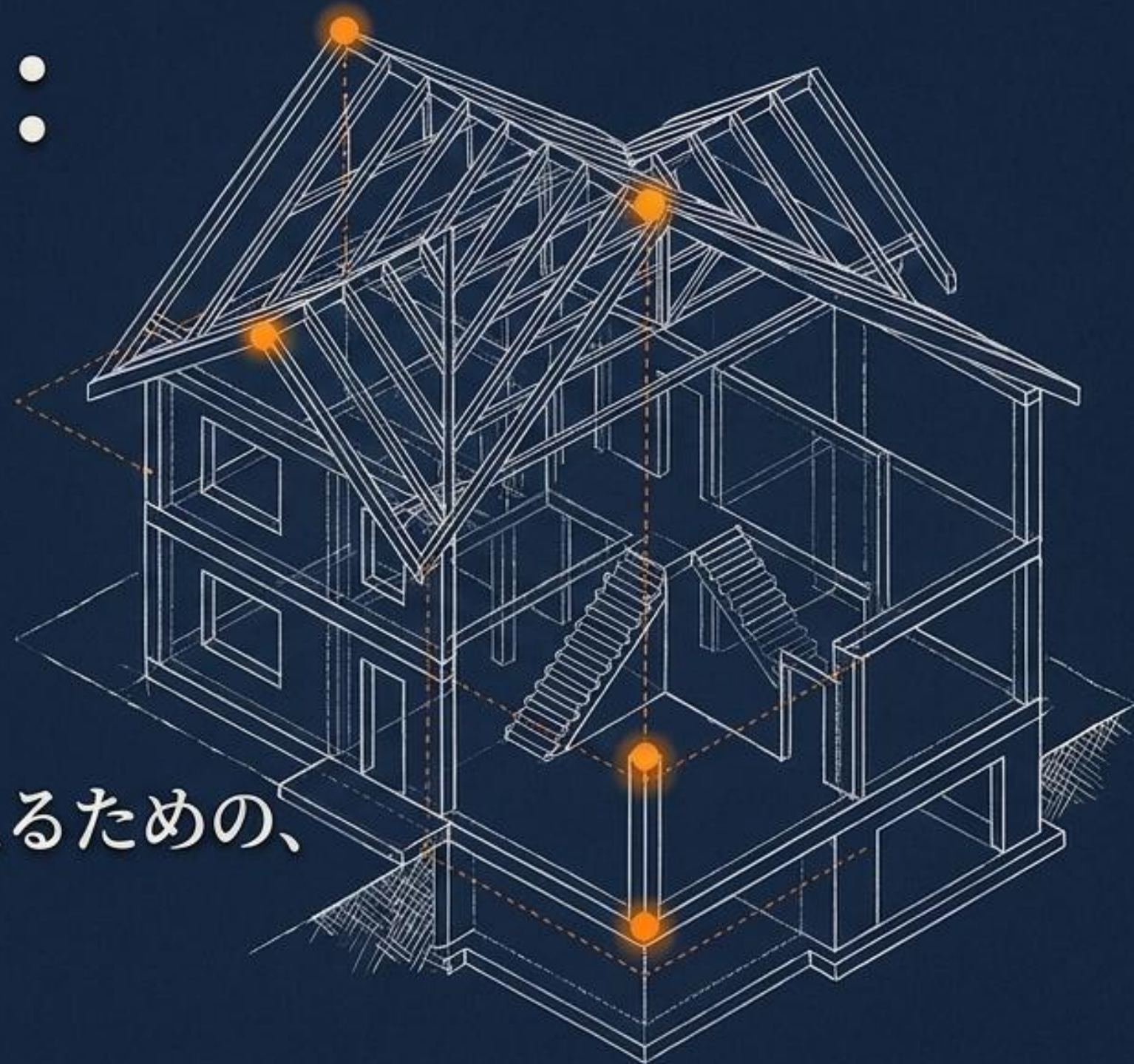
Housing Prices Competition for Kaggle Learn Users

Apply what you learned in the Machine Learning course on Kaggle Learn along...

Getting Started - 5024 Teams - Ongoing

限界突破の設計図： Ames Housing コンペティション 実践録

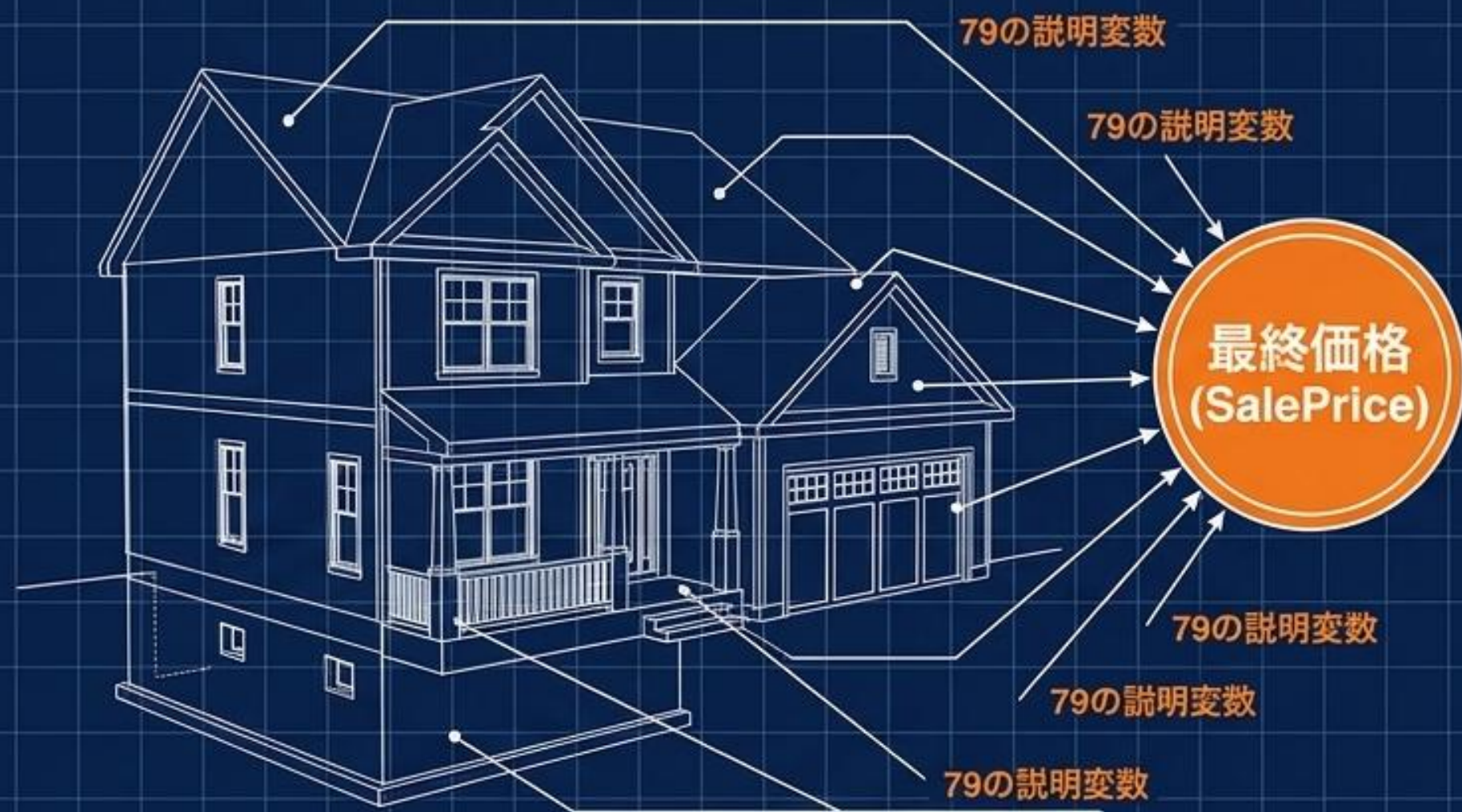
Kaggle初心者が12,000の壁を越えるための、
仮説検証と失敗のケーススタディ



プロジェクトの目的と評価基準

Target: アイオワ州エイズの住宅価格 (SalePrice) の予測。

Materials: 地下室の高さから鉄道への近さまで、79個の説明変数を使用。

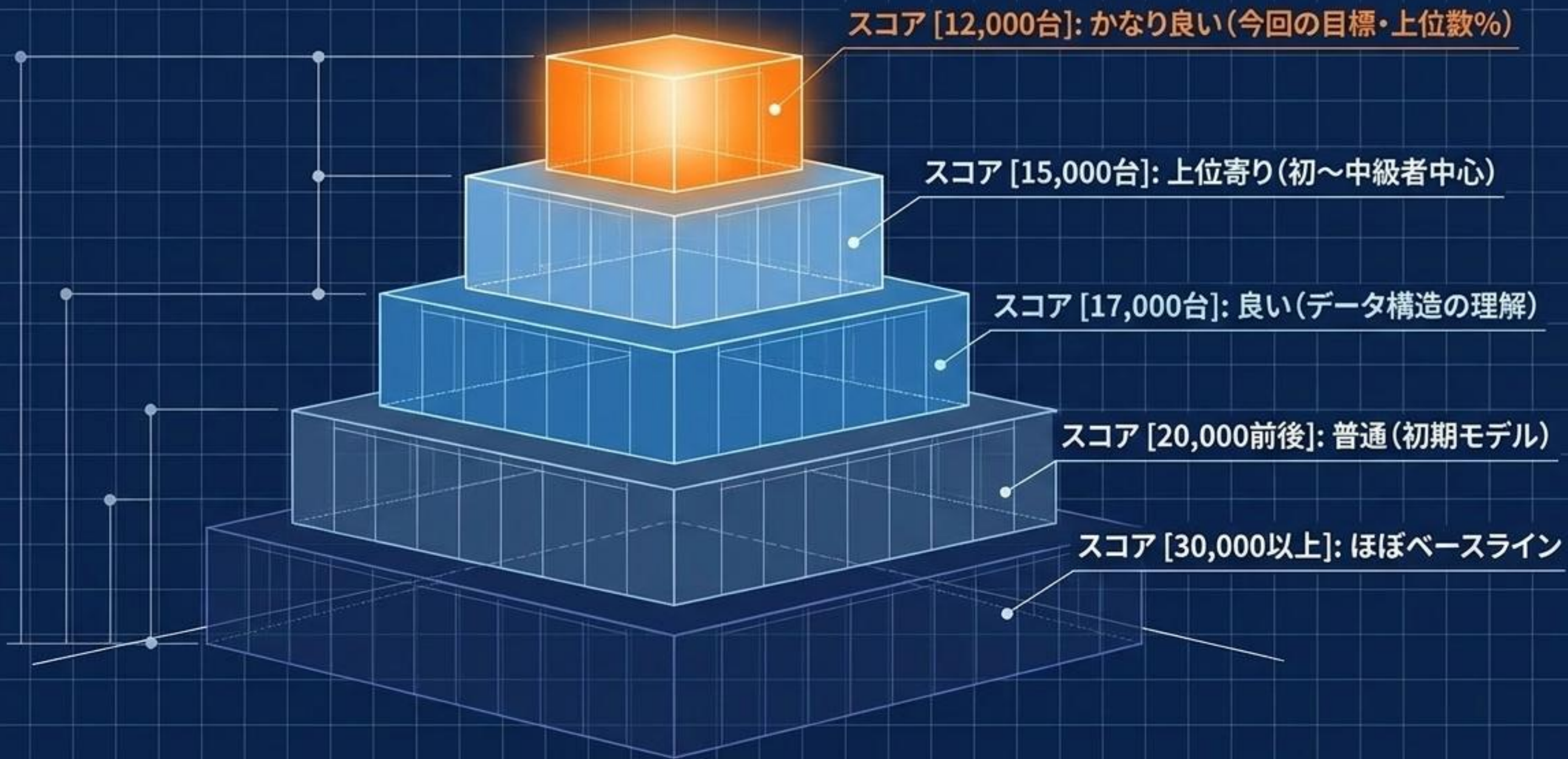


Metric: 予測値の対数と実際の販売価格の二乗平均平方根誤差 (RMSE)。

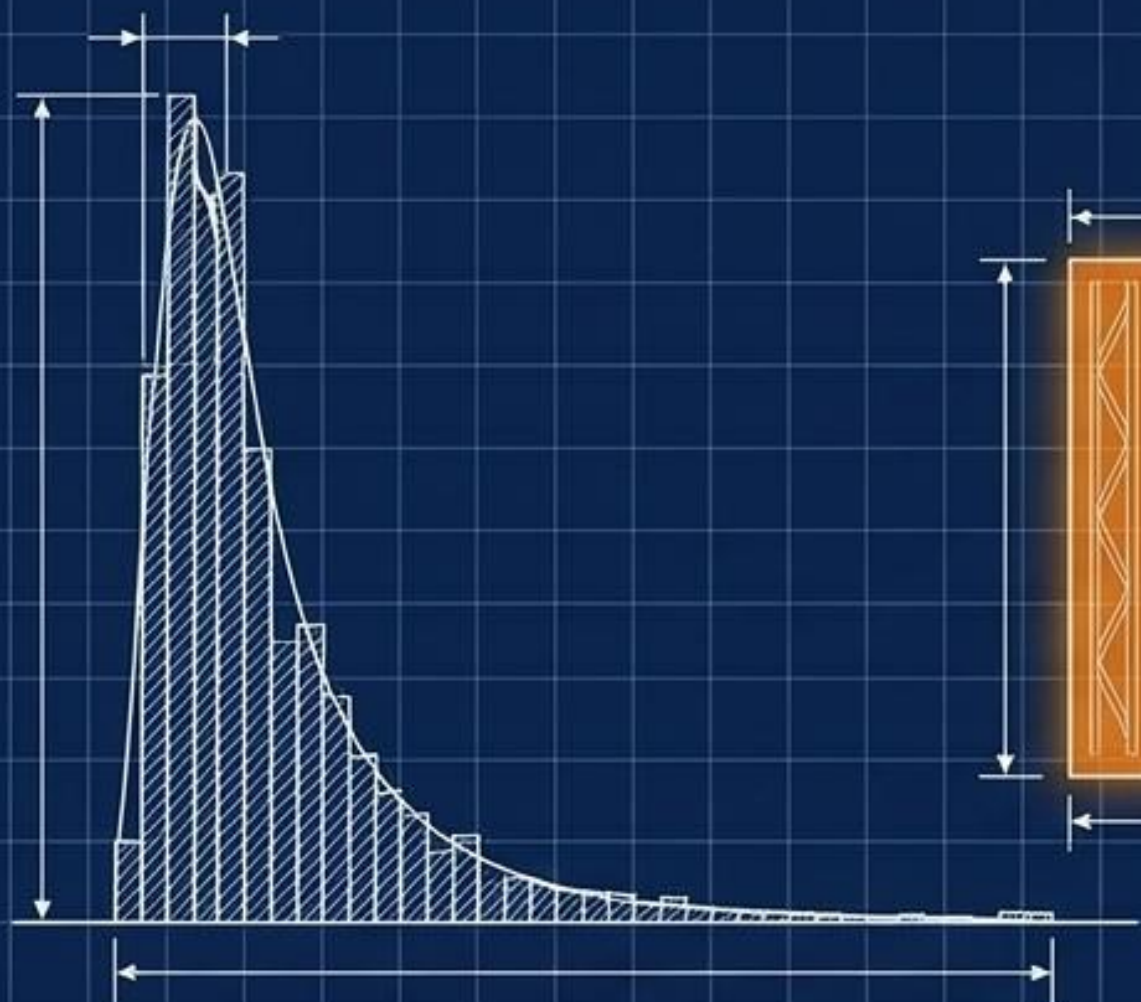
Rule: 高価な住宅も安価な住宅も、予測誤差が結果に等しく影響するように対数化が必須。

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (X_i - x_0)^2},$$

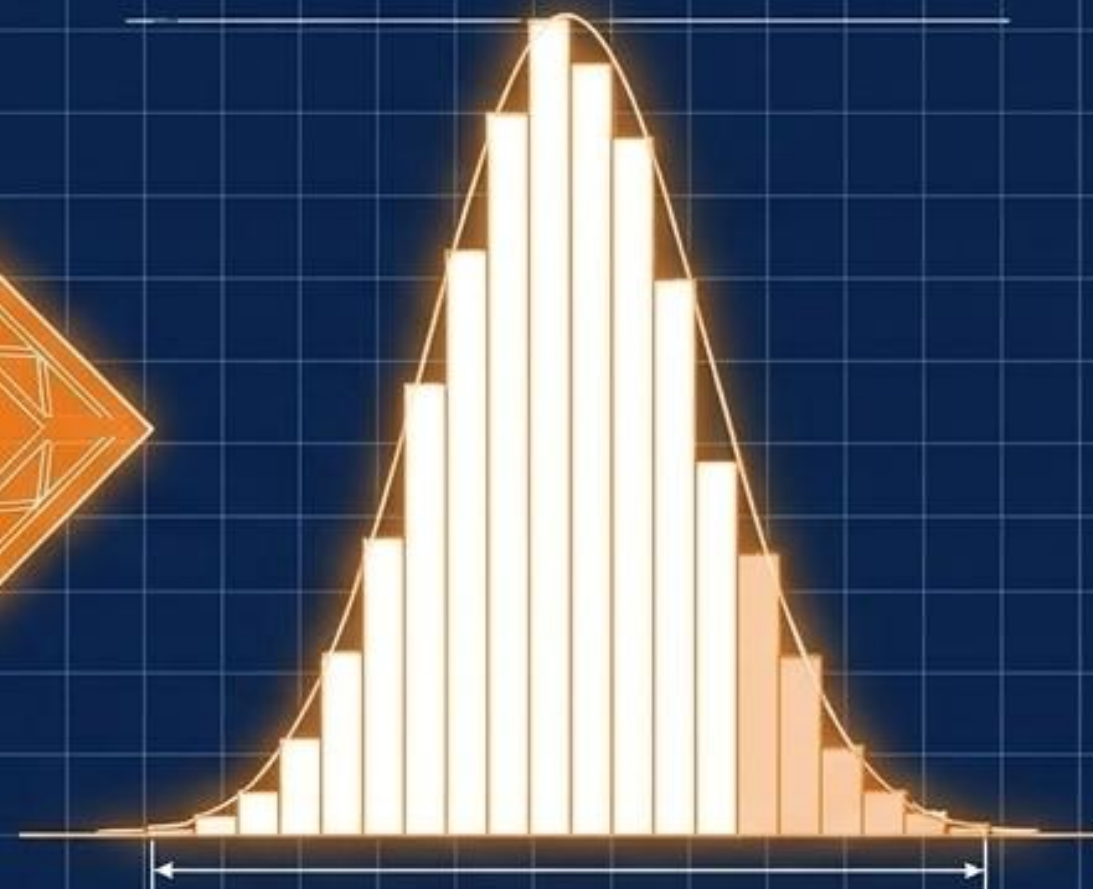
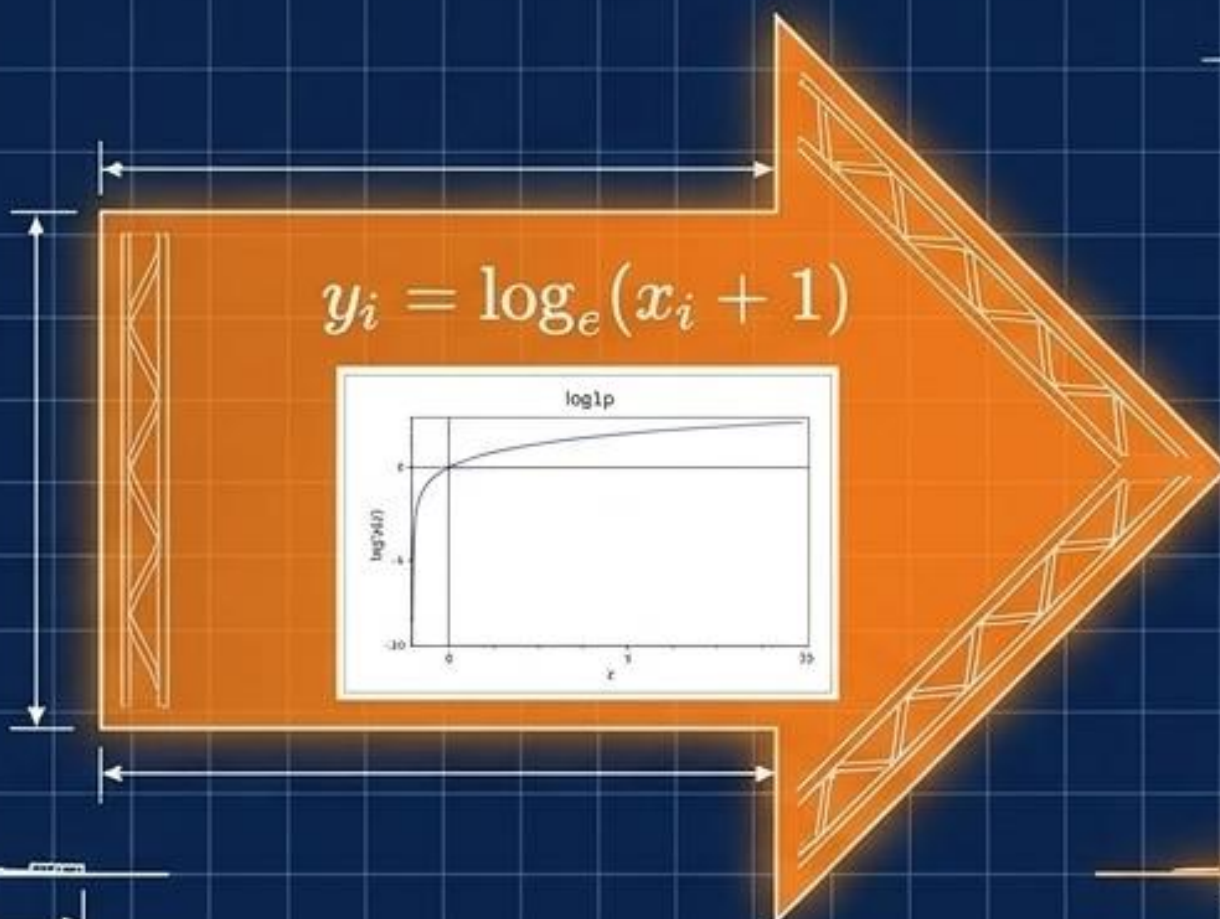
スコア分布と現在地：目指すは「12,000」の壁



The Target Trap : 右に歪む価格分布の補正



Before: 実際のSalePrice



After: SalePrice_log

現象: 少数存在する高価格帯の物件が、分布を右に大きく歪ませている。

必須アクション: 上位陣は例外なく `SalePrice_log = np.log1p(SalePrice)` を適用。

警告: これを怠ると、高額物件の誤差に引っ張られ、RMSEスコアが一気に悪化する。

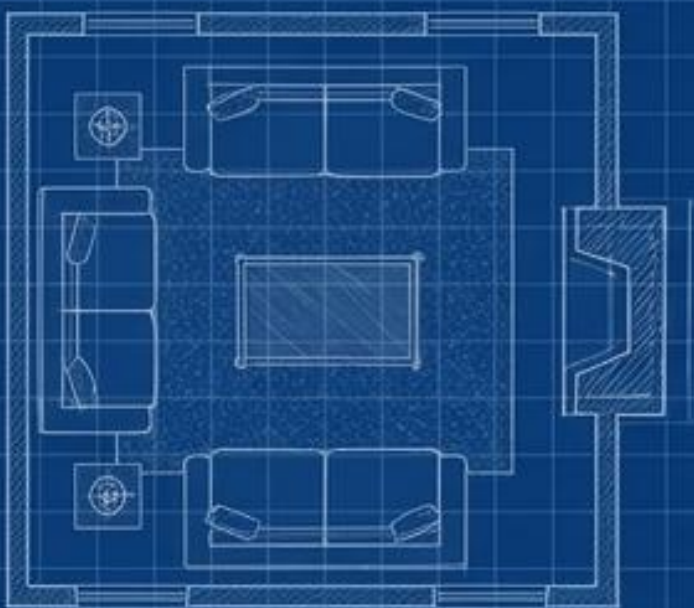
欠損値のパラダイムシフト：「欠損 ≠ 不明」

3 raw dataframe

Feature (欠損率)	Data Status	Real World Meaning	Action
PoolQC (99.5%)	NaN	プールが存在しない	None に置換
Alley (93.8%)	NaN	路地アクセスがない	None に置換
Fence (80.8%)	NaN	フェンスがない	None に置換
GarageArea	NaN	ガレージがない	0 に置換

Golden Rule: Rule: 上位解法は欠損を雑に捨てない。
「意味を持たせて補完」することで、情報量を最大化する。

The Feature Architecture: 価格を決定づける4つの柱



Quality (品質)

OverallQual
ExterQual
KitchenQual

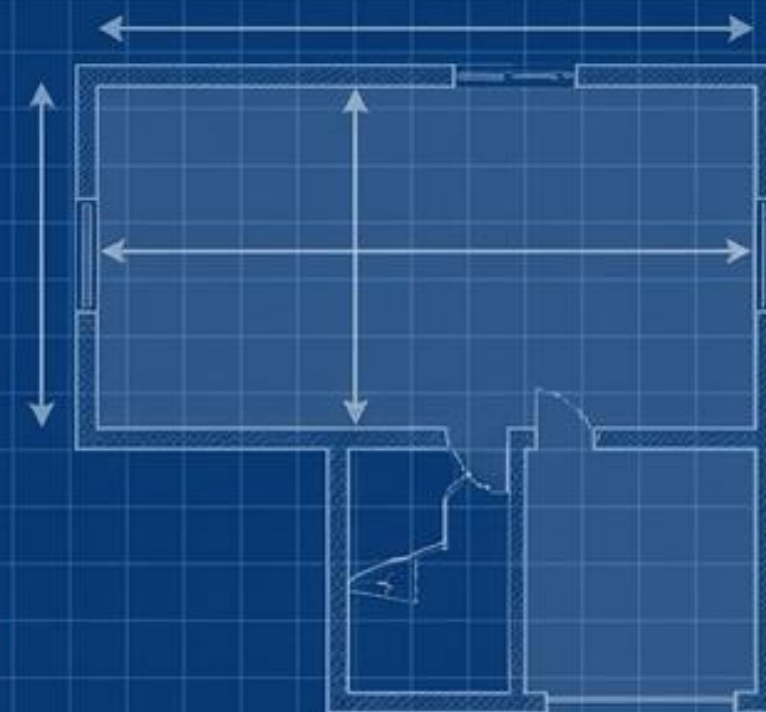


家全体の質。1~10評価の
OverallQual は全特徴量中
トップクラスの相関。

Space (広さ)

GrLivArea
TotalBsmntSF
GarageArea

広さ。地上居住面積
(GrLivArea) は価格とほぼ
線形関係。



Time (時間・歴史)

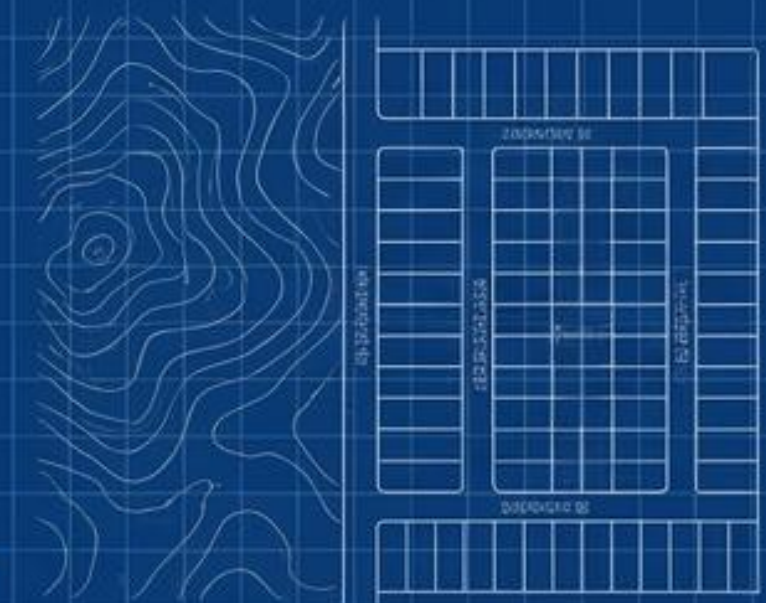
YearBuilt
YearRemodAdd
YrSold

築年数と改築。新しいほど
高く、改築が価格を押し上
げる。

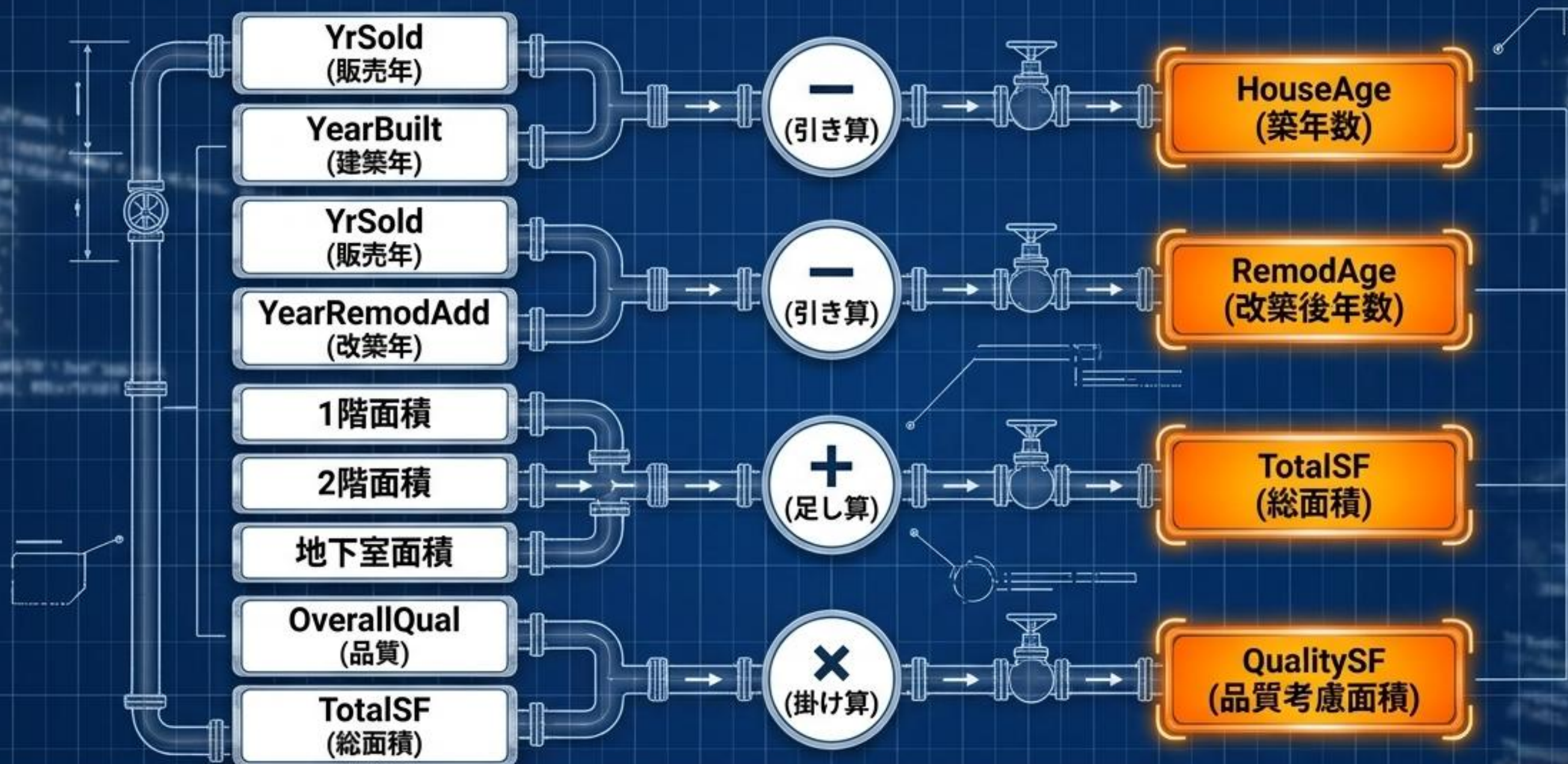
Location (立地)

Neighborhood

地区名 (Neighborhood)。
立地の価値を最もよく表す
最強のカテゴリ変数。



派生特徴量の錬成：既存データを「意味のある指標」へ



Takeaway: 「年」そのものより「経過年数」が効く。
元データを組み合わせることで、機械学習モデルが価格差を捉えやすくなる。

The Golden Baseline：定石による地盤固め

- ✓ Target: $\log_{1p}(\text{SalePrice})$
- ✓ Model: LightGBM (LGBMRegressor) / KFold CV
- ✓ Missing Values: None や 0 への意味的補完
- ✓ Outliers: GrLivArea > 4000 かつ低価格の異常値を除去
- ✓ Categoricals: そのまま全使用 (One-Hot Encoding)

Score:
14,638

Status: 「15,000台の壁」は突破。
しかし、目標の12,000にはまだ遠い。

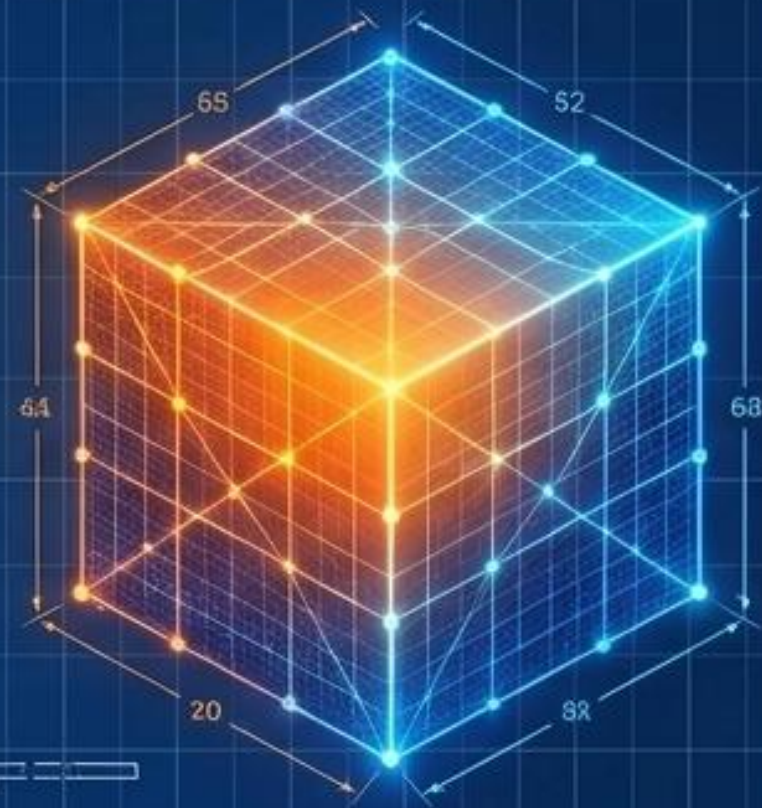
カテゴリ変数の壁と CatBoost の躍進

LightGBM (One-Hot)



One-Hotエンコーディングは Neighborhood など
を「独立したフラグ」にしてしまい、
カテゴリ間の関係性や順序を学べない。

CatBoost (Native Categorical)

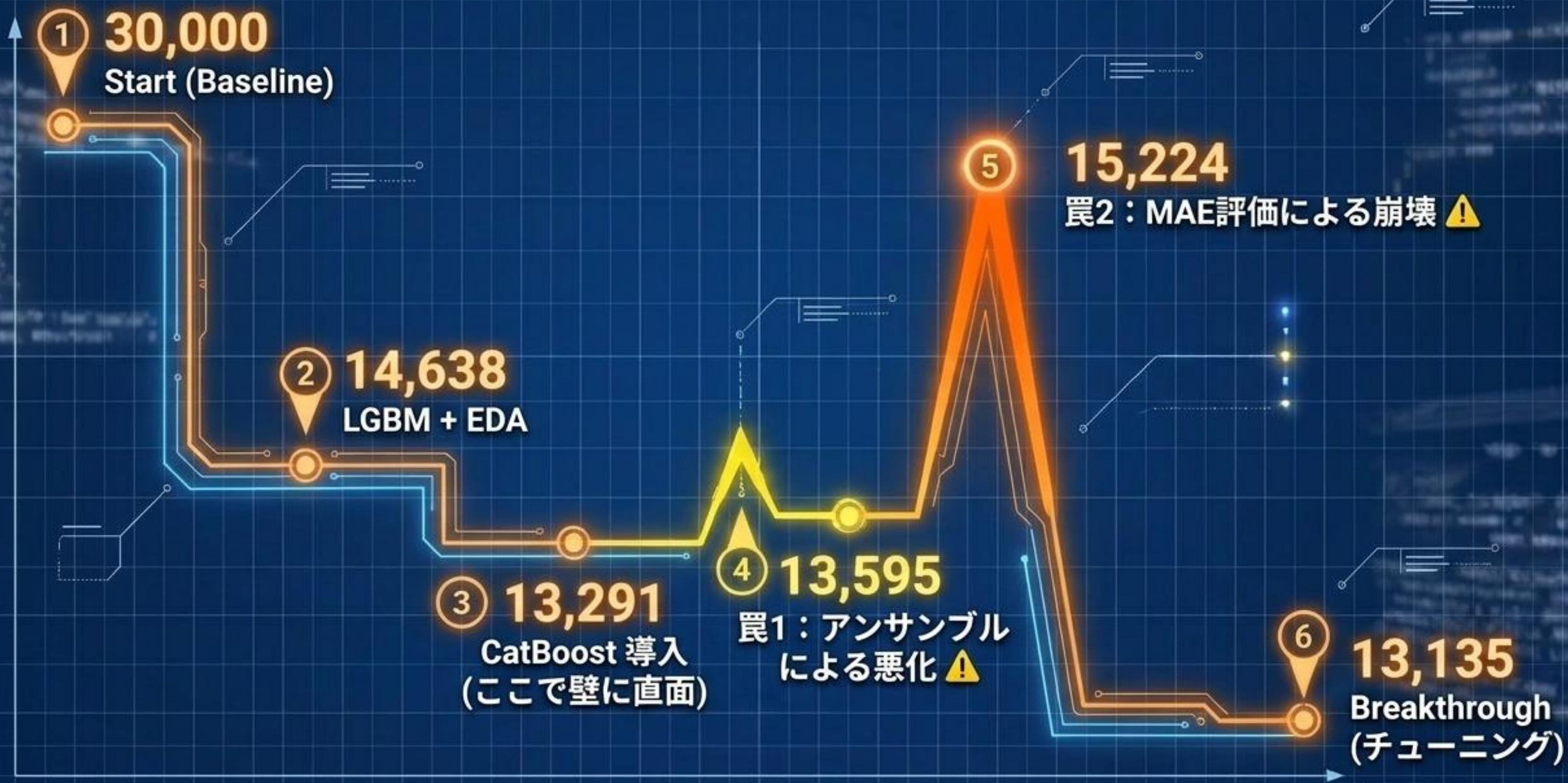


カテゴリ変数をそのまま賢く扱えるCatBoostを導入。

New Score: 13,291

(実行時間15分。正攻法として非常に強力なスコアへ到達)

The Timeline of Volatility : スコア乱高下の軌跡



Trap 1: ブレンド (アンサンブル) の幻想



The Incident:

LGBM(14638)とCatBoost(13291)を
0.5:0.5 でブレンド

→ スコアは 13,595 に悪化。

The Diagnostic:

ブレンドが効くのは「互いに違う場所
を間違える」時のみ。

今回は LGBM が CatBoost と同じ方
向で「より大きく間違えていた」。

結果として、CatBoostの良い予測を
悪い予測で薄めてしまった。

Trap 2: MAE学習による予測の崩壊

RMSEの性質



RMSE: 誤差を「二乗」するため、高額物件の大きな予測外しを強く罰する。

MAEの性質



MAE: 外れ値を気にせず中央値に寄りやすい。結果として、コンペ評価で重要な「高額物件」の予測を意図的に外してしまう。



The Incident: 損失関数をMAEに変更
→ スコアは一気に 15,224 へ急落。

The Breakthrough: アーキテクチャの再調整

CatBoost Depth



CV Score Comparison

Depth	CV RMSE
[depth=7]	23186.08 (Best)
[depth=8]	23898.19
[depth=9]	24820.55
[depth=10]	24696.64

Action 1: RMSE学習への回帰 (loss_function='RMSE')

Action 2: CatBoostのDepthチューニング

Result: シンプルなモデル深さの最適化により、スコアは **13,135** へと最高値を更新。
複雑な手法に頼る前の「基本の徹底」が鍵。

The Final Blueprint : 12,000を切るための最終戦略



The Next Step: Target Encoding (Neighborhood)

Why it works:

- 同じ地区 (Neighborhood) の「平均価格」を特徴量としてモデルに与える。
- 単なるカテゴリラベル (One-Hot) ではなく、具体的な「価格の相場感」を直接学習させることで、精度が一段階引き上がる。

Caution:

- 必ずCV (交差検証) の中でエンコードを行い、過学習 (データリーク) を防ぐこと。

The Architect's Summary : Ames攻略の4つの黄金則

1

Rule 1: Transform the Target

SalePrice は必ず $\log_{10}$ で対数化せよ。

2

Rule 2: Translate Missing Data

欠損値(NaN)は「存在しない」という強力な情報である。意味を持たせて補完せよ。

3

Rule 3: Beware the Blend

劣ったモデルとの単純なブレンドは、優れた予測を汚染する。

4

Rule 4: Respect the Metric

RMSE評価のコンペでは、高額物件の誤差を無視するMAE学習は命取りになる。

Now, build your own breakthrough. (さあ、あなた自身の限界突破を設計しよう)