



# ANÁLISE EM ÁREAS DE PONDERAÇÃO

# SUMÁRIO

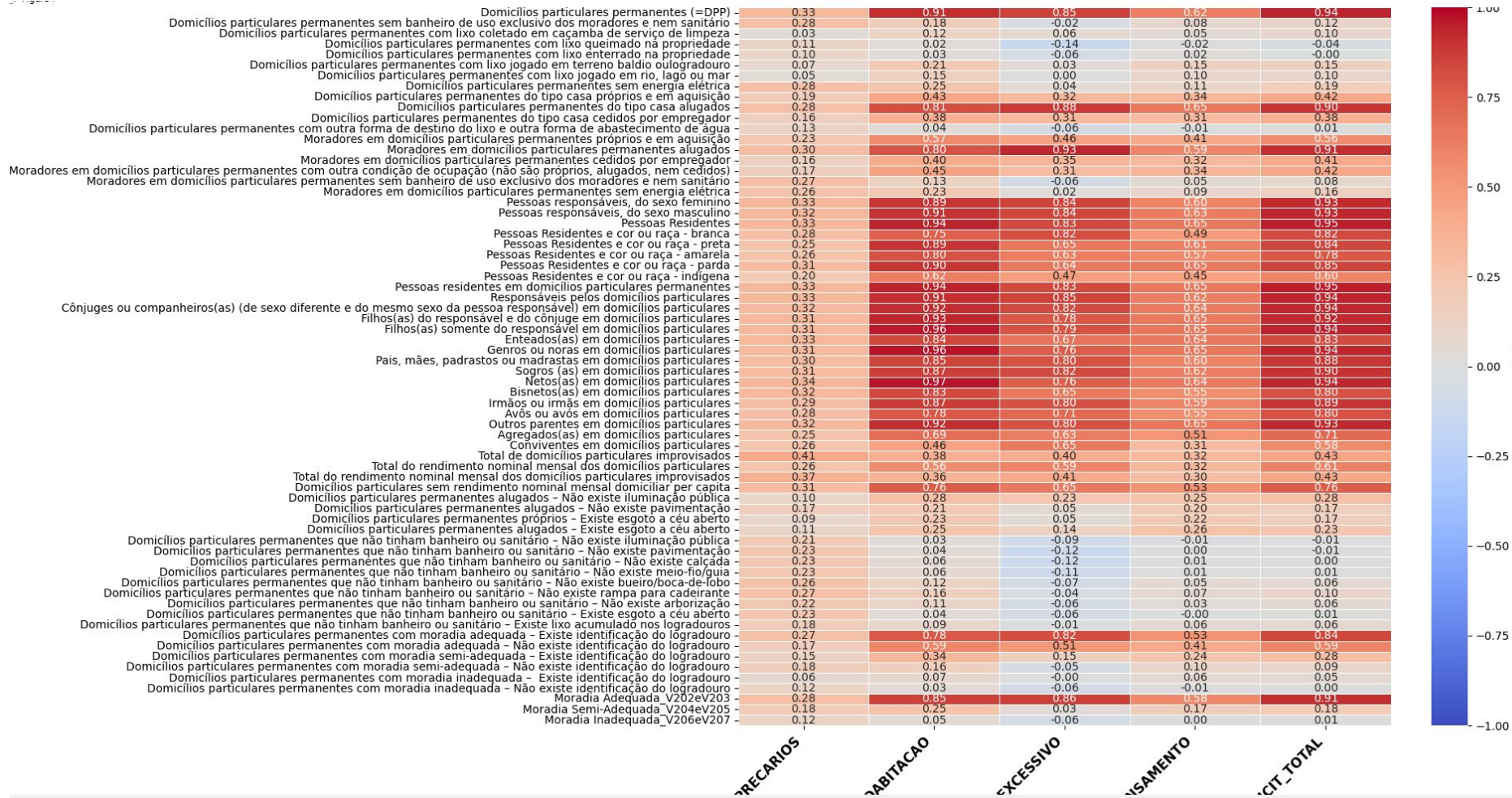


1. Alterações nome das entradas: refatoração mapa de calor, variáveis mais importantes, variáveis com mais zeros;
2. Testes com otimização na Random Forest: utilização de novos hiperparâmetros, maior explicabilidade para os resultados anteriores, mais referências para a elaboração do artigo;
3. Contribuição e dispersão individual das áreas de ponderação por instância através de mapa Beeswarm
4. Quantificação no impacto marginal de cada variável no desempenho através de permutação de importância;
5. Mapeamento espacial dos erros: identificação das AP's que possuem maior erro e possibilidade de análise de erros por padrões geográficos;
6. Sobre desagregação e próximos passos

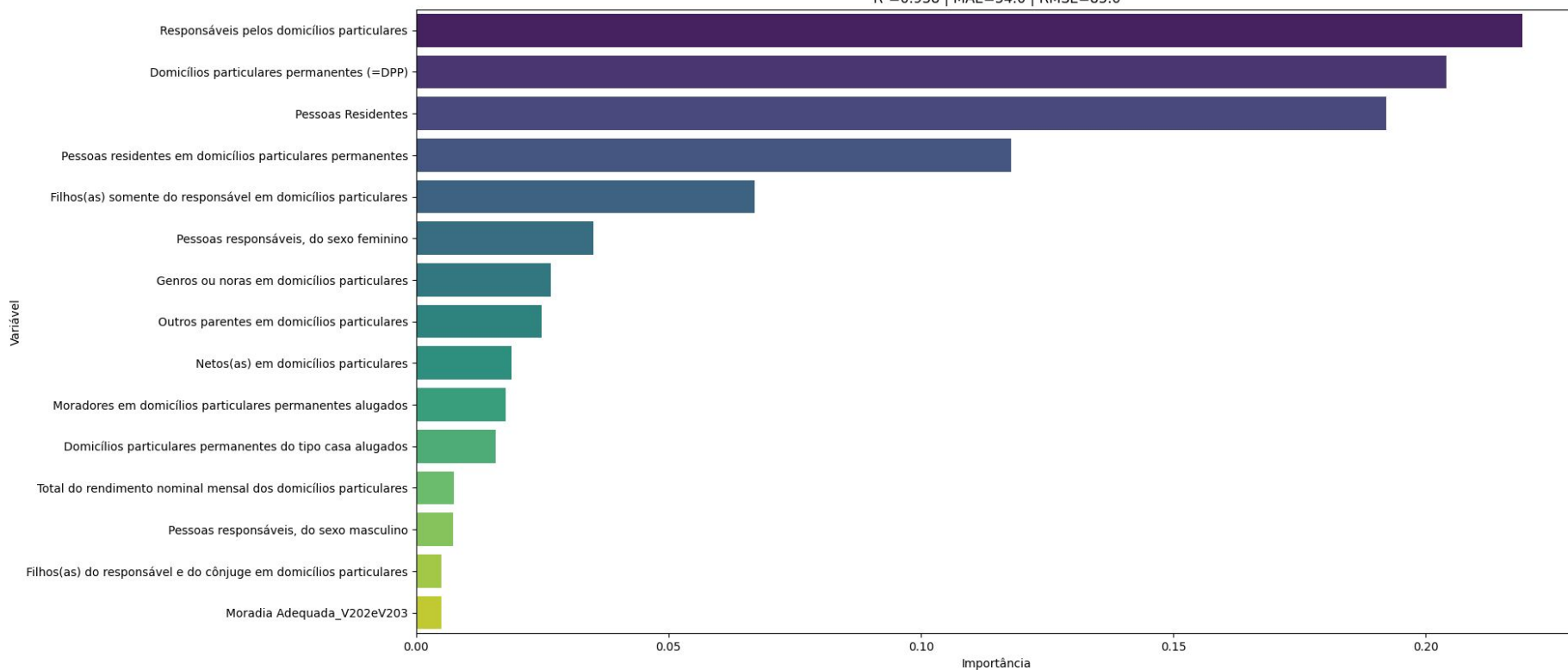
# 1. Alterações nome da entradas

Refatoração mapa de calor, variáveis mais importantes, variáveis com mais zeros;

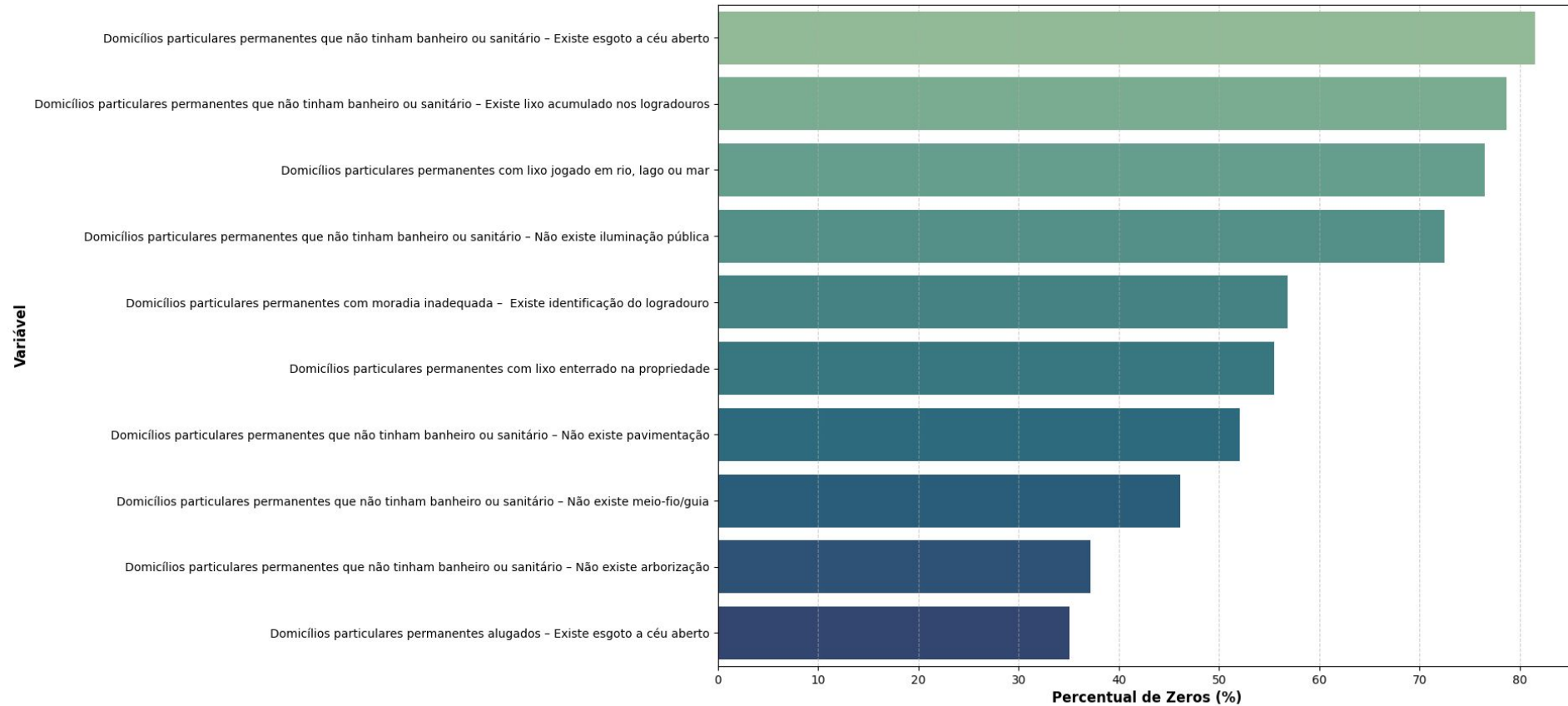
---



Importância das variáveis para DEFICIT\_TOTAL  
 $R^2=0.938$  | MAE=54.0 | RMSE=85.0



**Top 10 Variáveis com Maior Percentual de Zeros**



## 2. Testes com otimização na Random Forest

Utilização de novos hiperparâmetros, maior explicabilidade para os resultados anteriores, mais referências para a elaboração do artigo;

---

# Testes com otimização

```
def objetivo_optuna(trial, X_train, y_train, target_name):  
    params = {  
        'n_estimators': trial.suggest_int('n_estimators', 100, 500), # num de arvores, s  
        'max_depth': trial.suggest_int('max_depth', None, 40), # controle de co  
        'min_samples_split': trial.suggest_int('min_samples_split', 2, 20), #num min de amo  
        'min_samples_leaf': trial.suggest_int('min_samples_leaf', 5, 10), #num min de moa  
        'max_features': trial.suggest_categorical('max_features', ['sqrt', 'log2', None]),  
        'bootstrap': trial.suggest_categorical('bootstrap', [True, False]), #sorteia amostr  
        #na arvore e outras nao, aumenta SIGNIFICAMENTE a diversidade das trees impedindo d  
        'random_state': RANDOM_STATE,  
        'n_jobs': -1 # Usa todos os cores disponíveis  
    }  
  
    modelo = RandomForestRegressor(**params)
```

Padrão biblioteca sklearn 1.3:

- n\_estimators = 100;
- max\_depth = None;
- min\_sample\_leaf = 1
- max\_features = default='sqrt'
- bootstrap = True
- random\_state = None
- n\_jobs = None (usa somente um core)

```
# CONFIGURAÇÕES  
N_TRIALS = 100 # Número de tentativas de otimização - numero de combinacoes dos parametros  
CV_FOLDS = 3 # cross validation - do dataset de treino estaremos repartindo em 4 e fazendo N trials ali de cima  
TEST_SIZE = 0.2 # tamanho dos dados para treino e teste (80/20)  
RANDOM_STATE = 42 #setta a ordem de randomização, padrao para reprodutividade dos testes
```

# Testes com otimização



<https://www.stat.berkeley.edu/~breiman/randomforest2001.pdf>

O desempenho da floresta aleatória depende da relação entre correlação e força

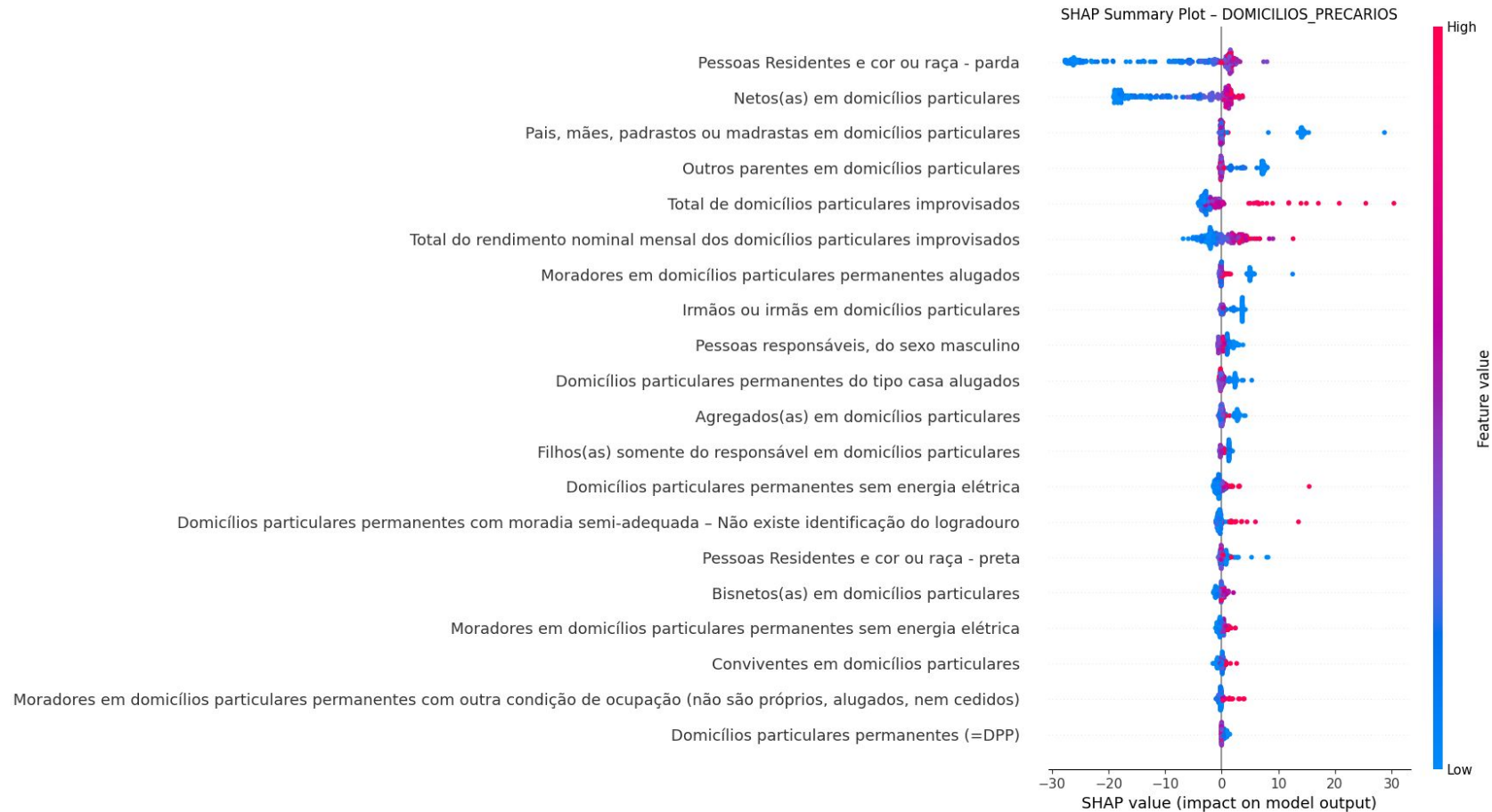
**Força:** Predição boa de cada árvore individualmente

**Correlação** entre árvores: quão parecida são as árvores, se todas as árvores cometem o mesmo erro a floresta é fraca

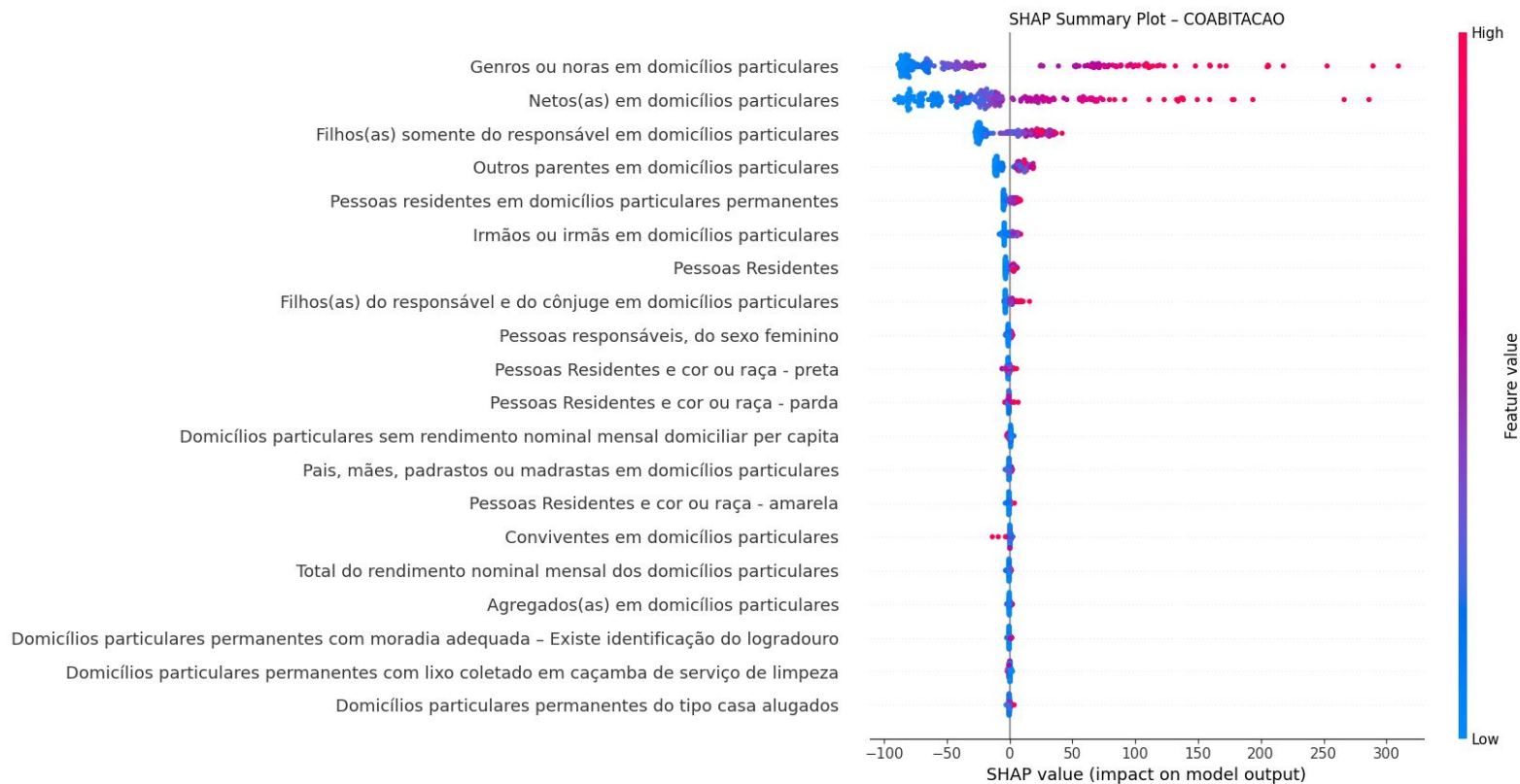
### 3. Contribuição e dispersão individual das áreas de ponderação por instância

Gráfico Beeswarm

---

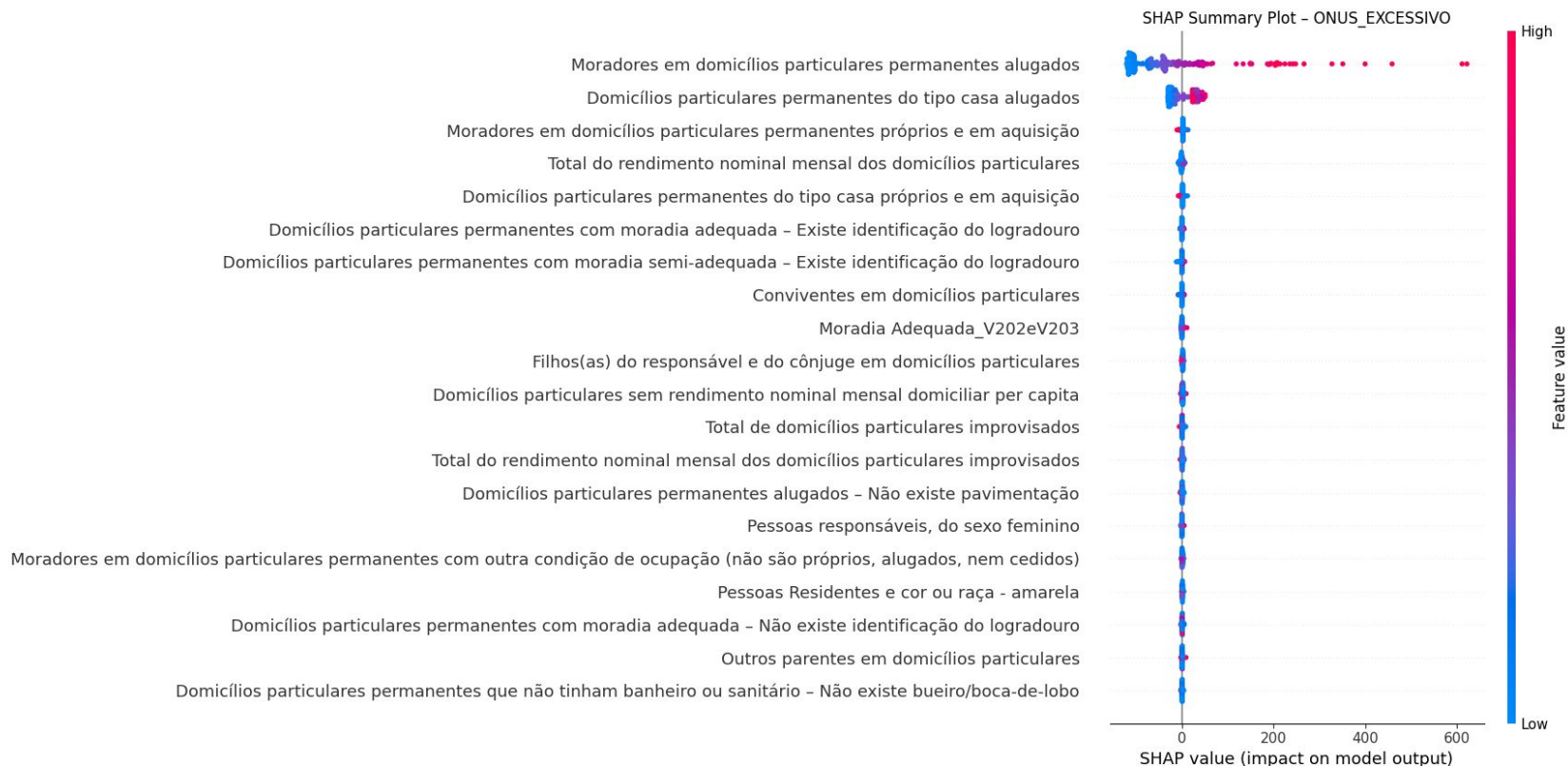


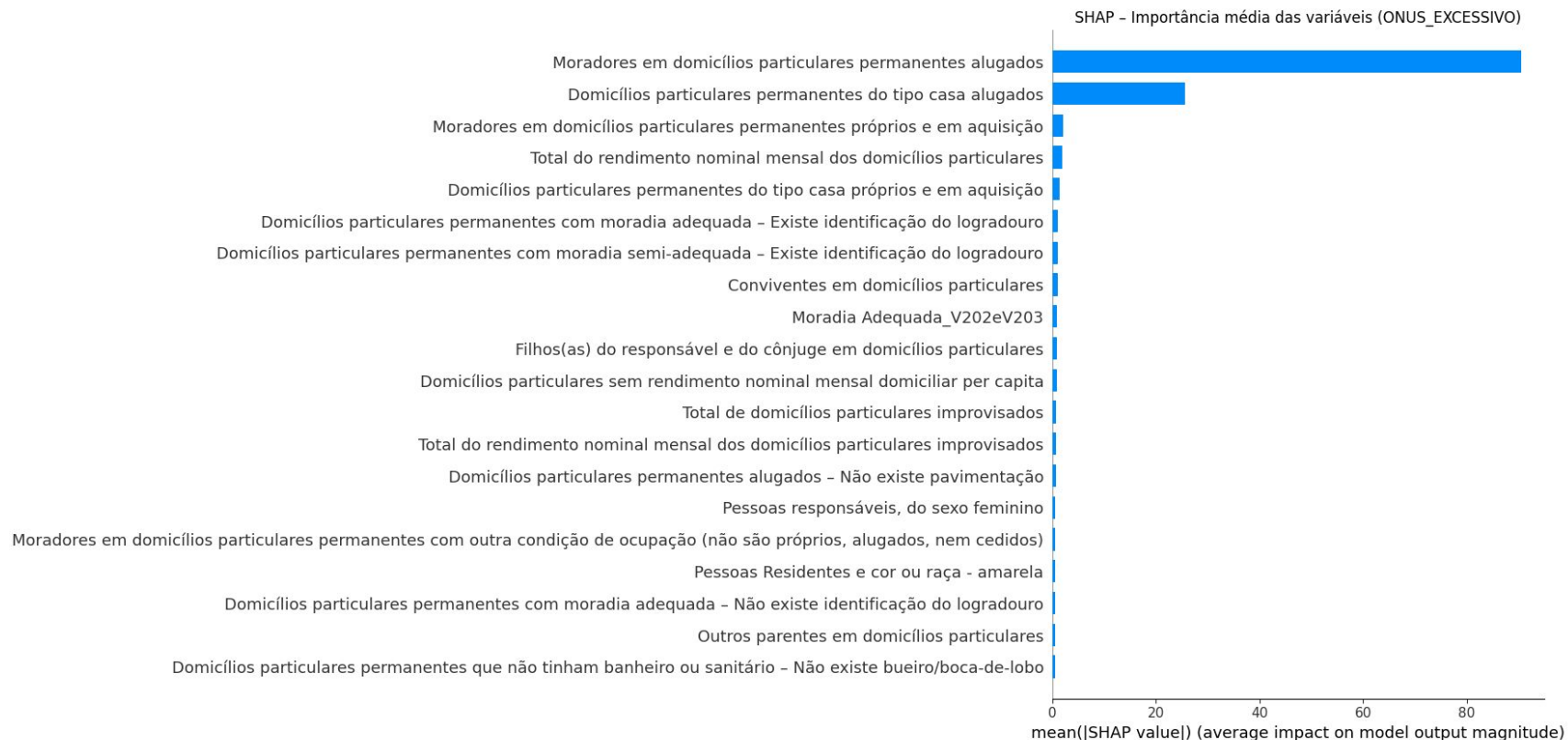


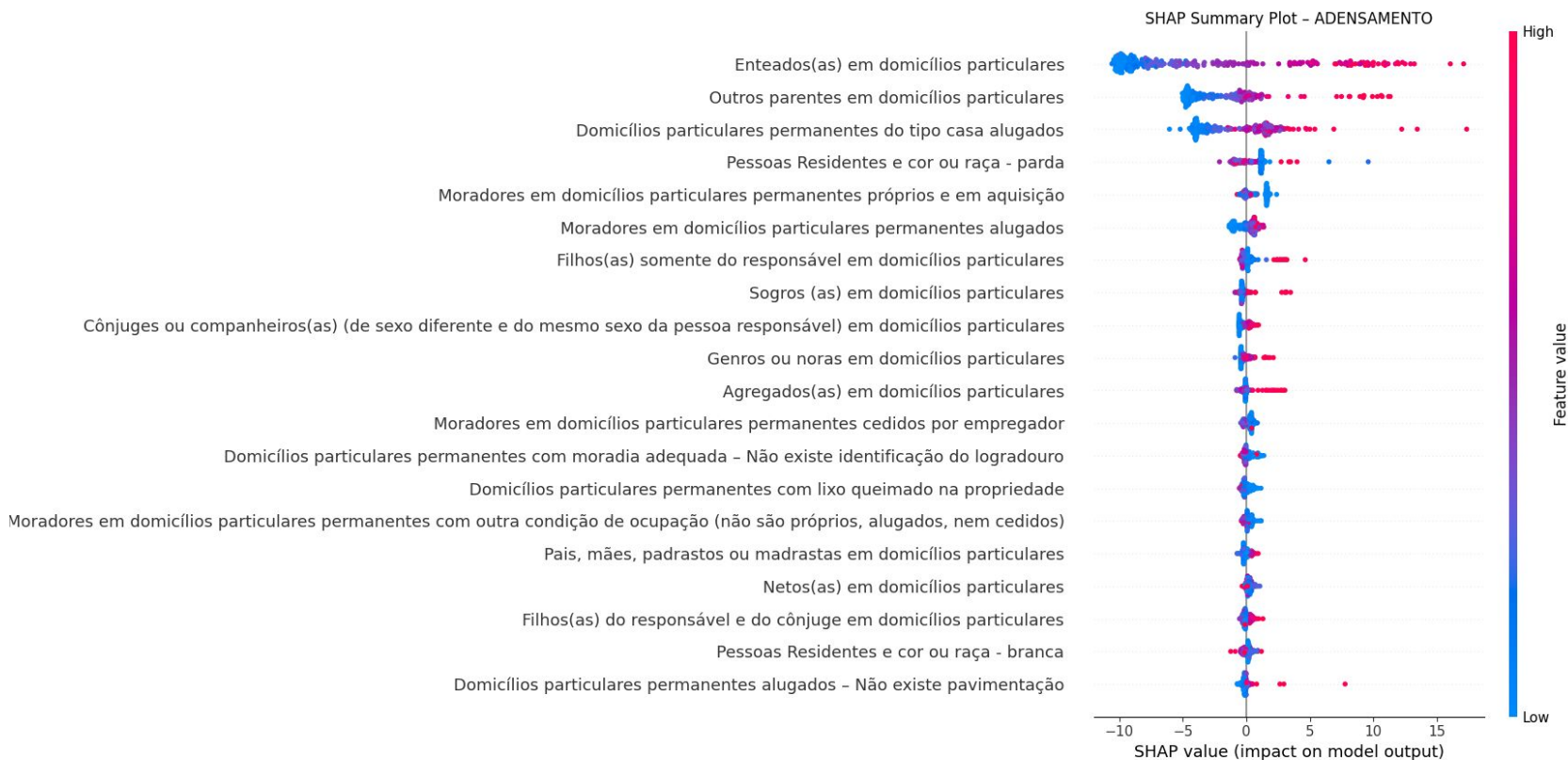


SHAP - Importância média das variáveis (COABITACAO)

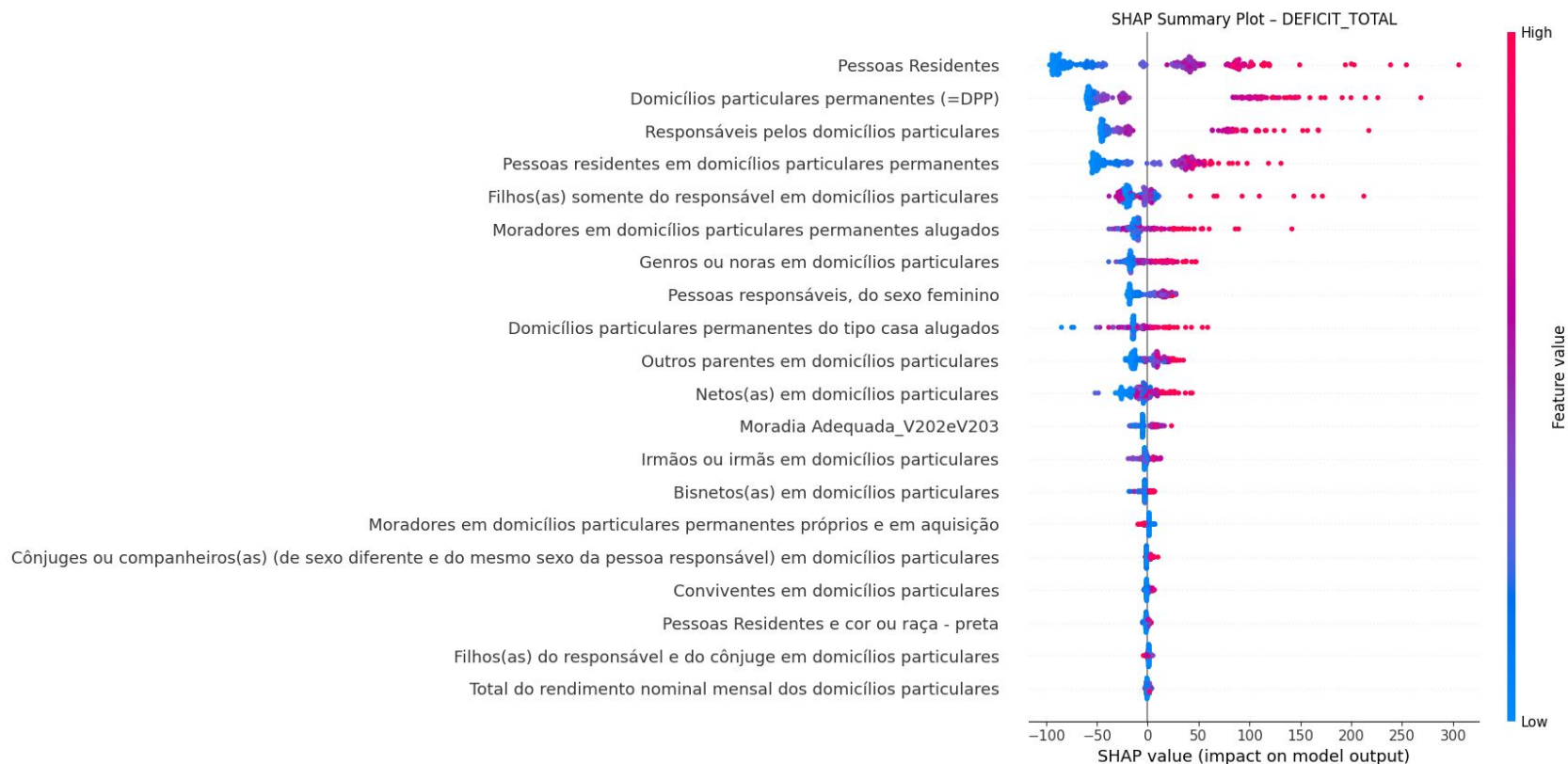










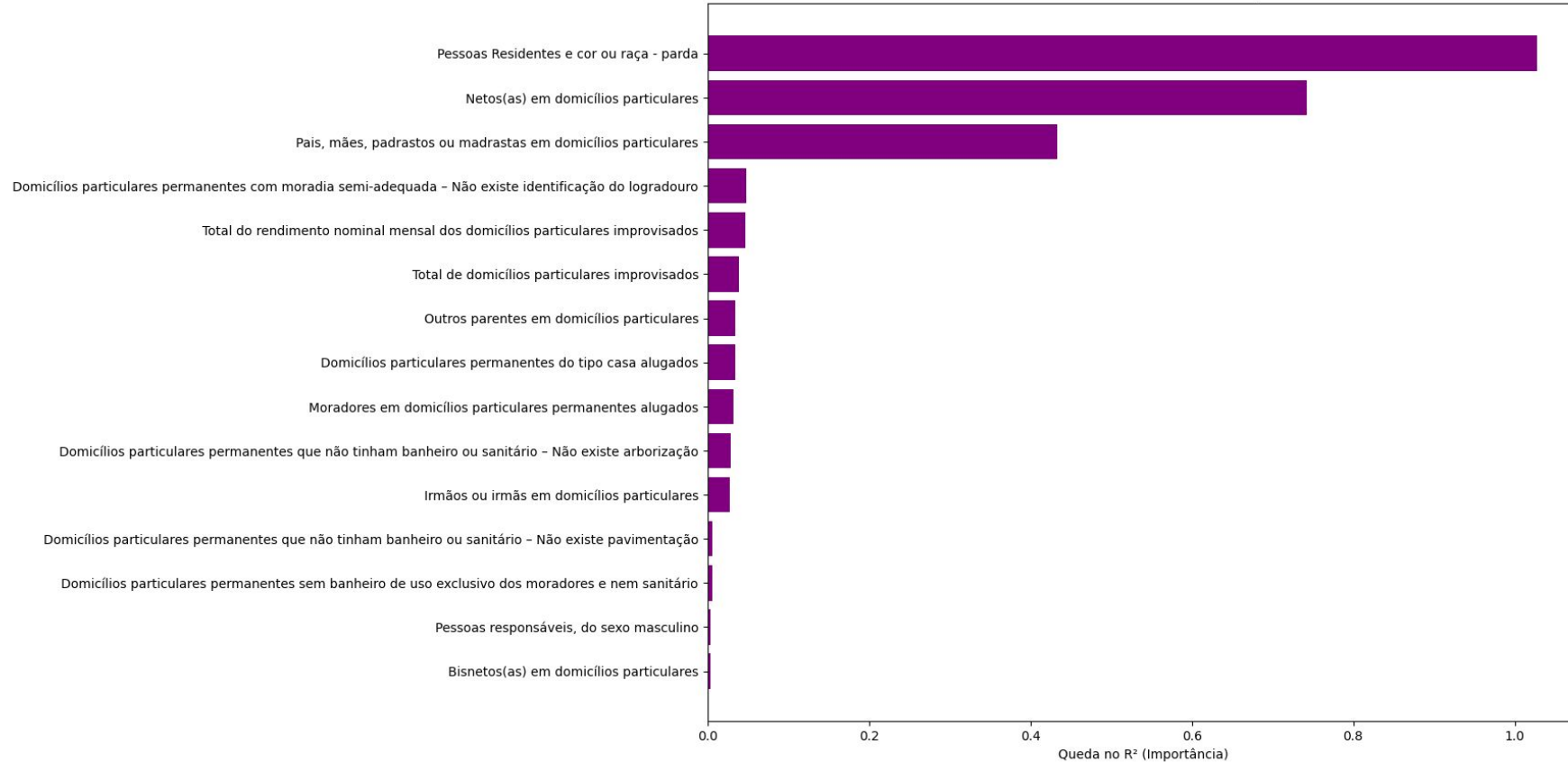




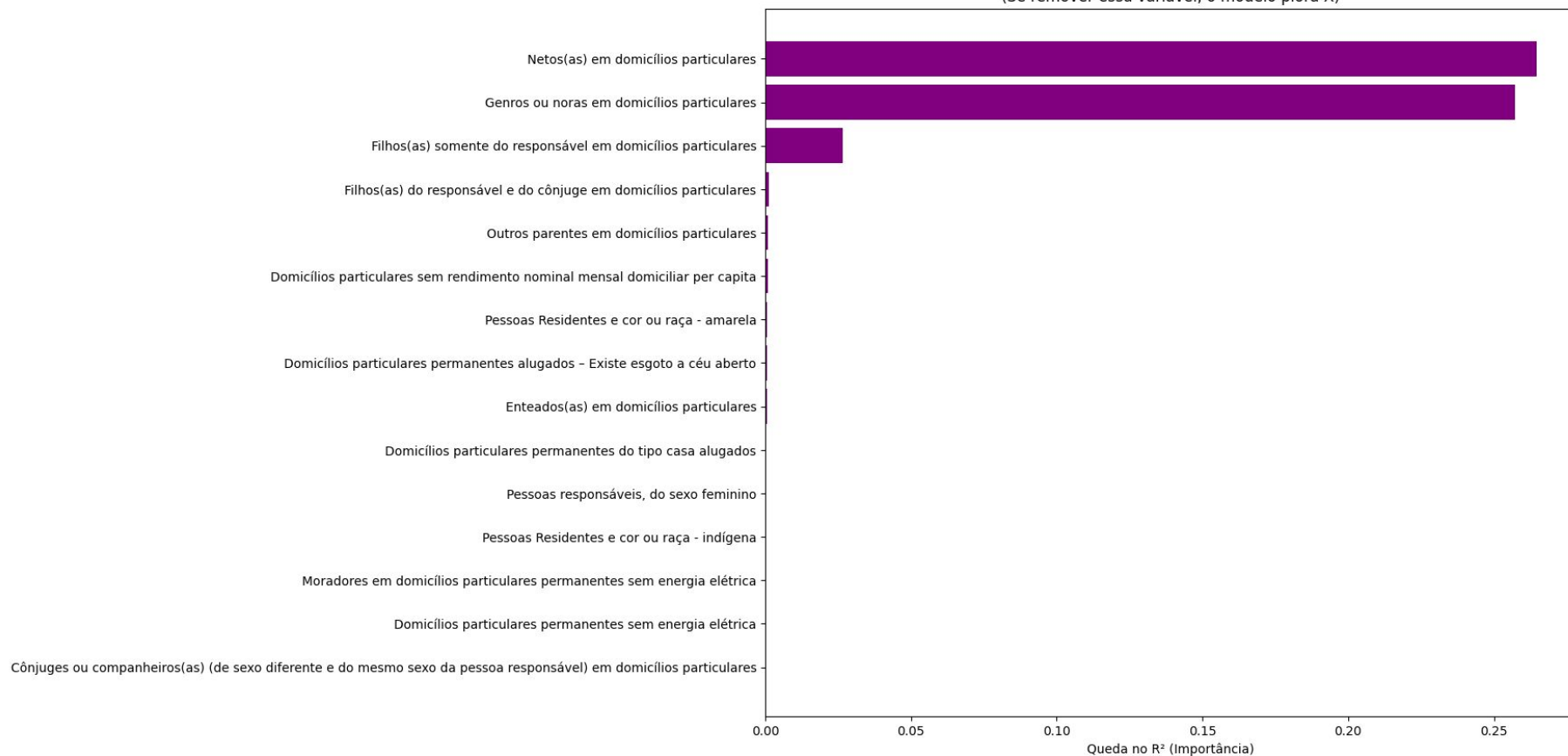
4. Quantificação no impacto marginal de cada variável no desempenho através de permutação de importância

---

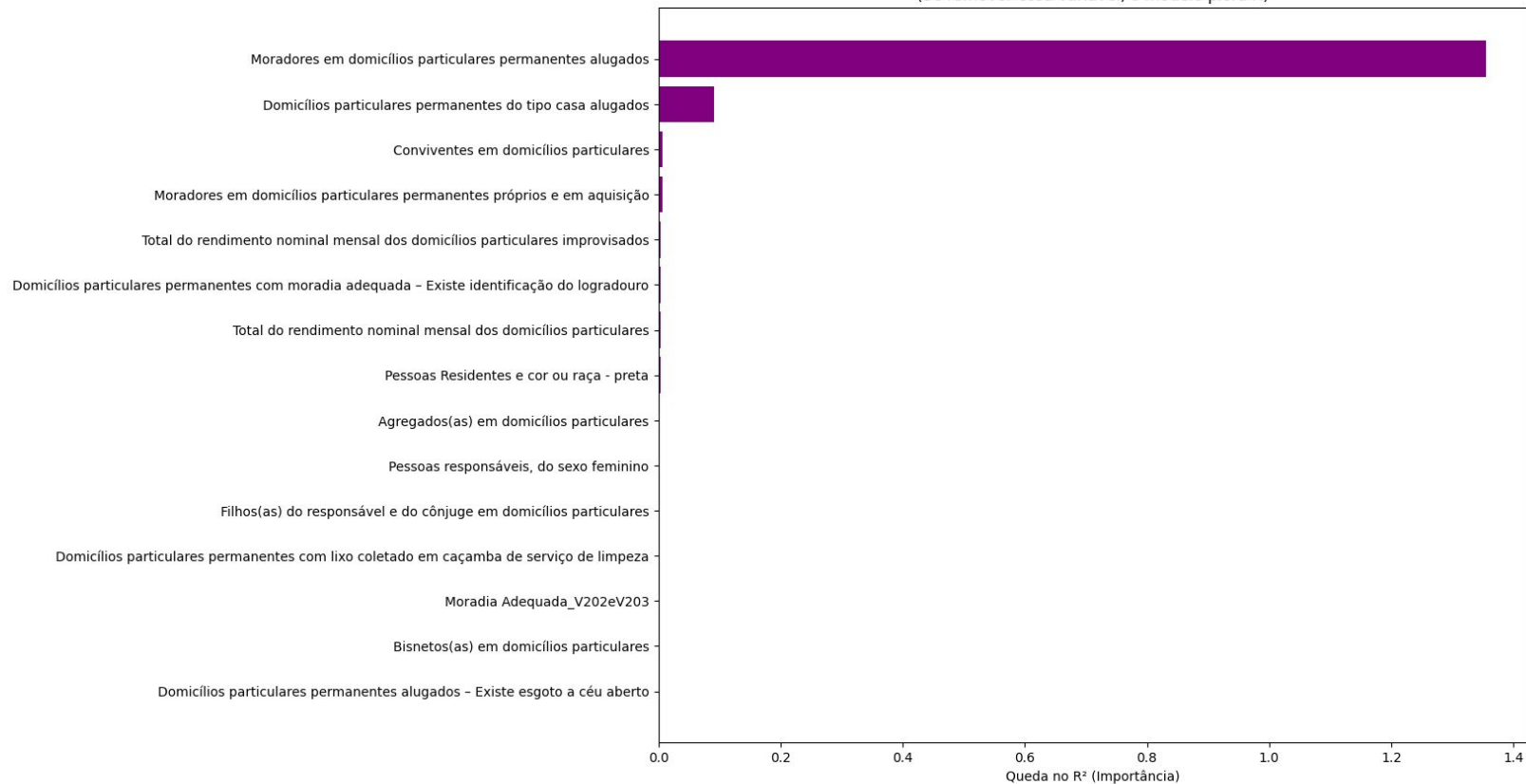
Permutation Importance - DOMICILIOS\_PRECARIOS  
(Se remover essa variável, o modelo piora X)



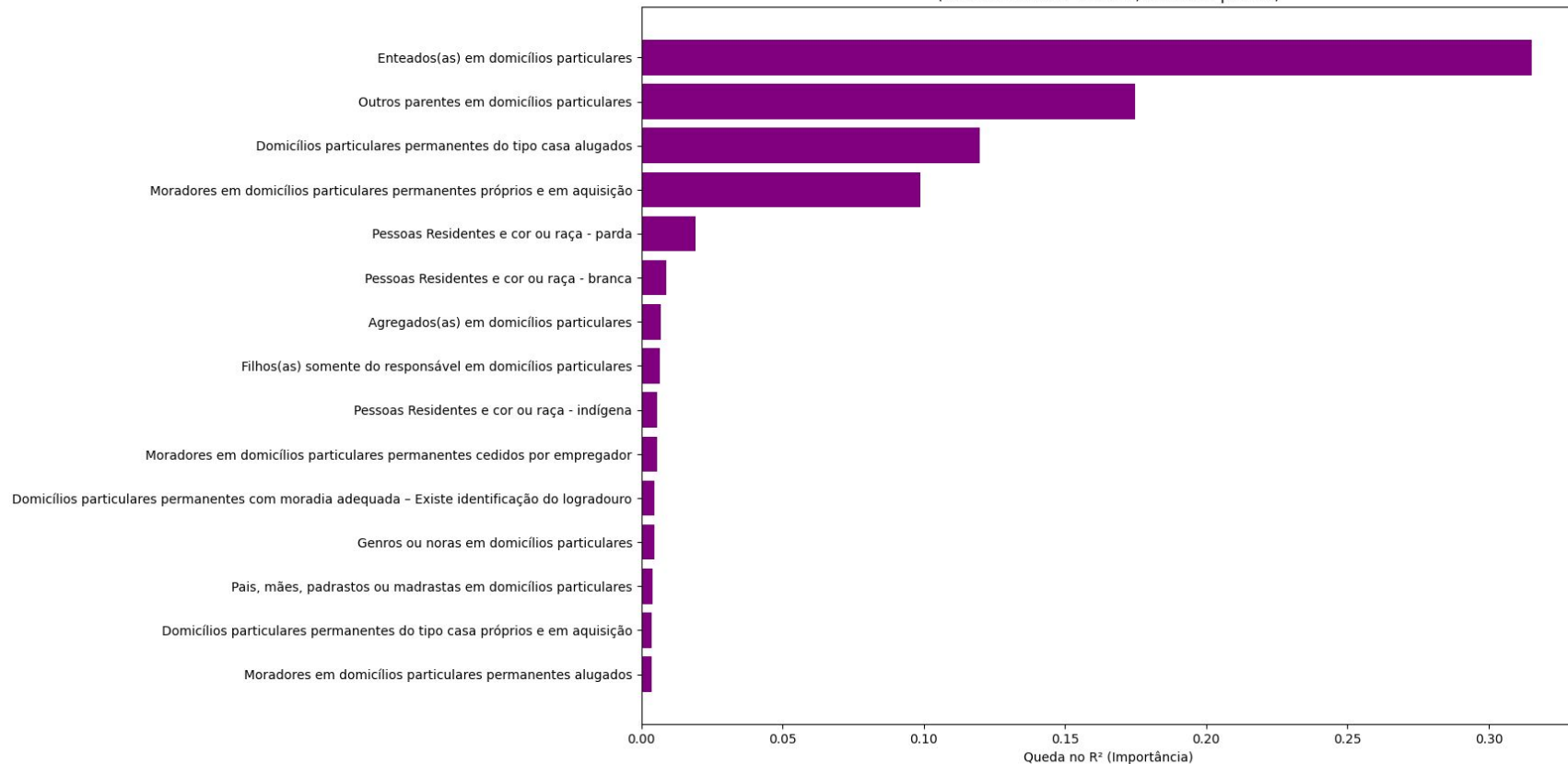
Permutation Importance - COABITACAO  
(Se remover essa variável, o modelo piora X)



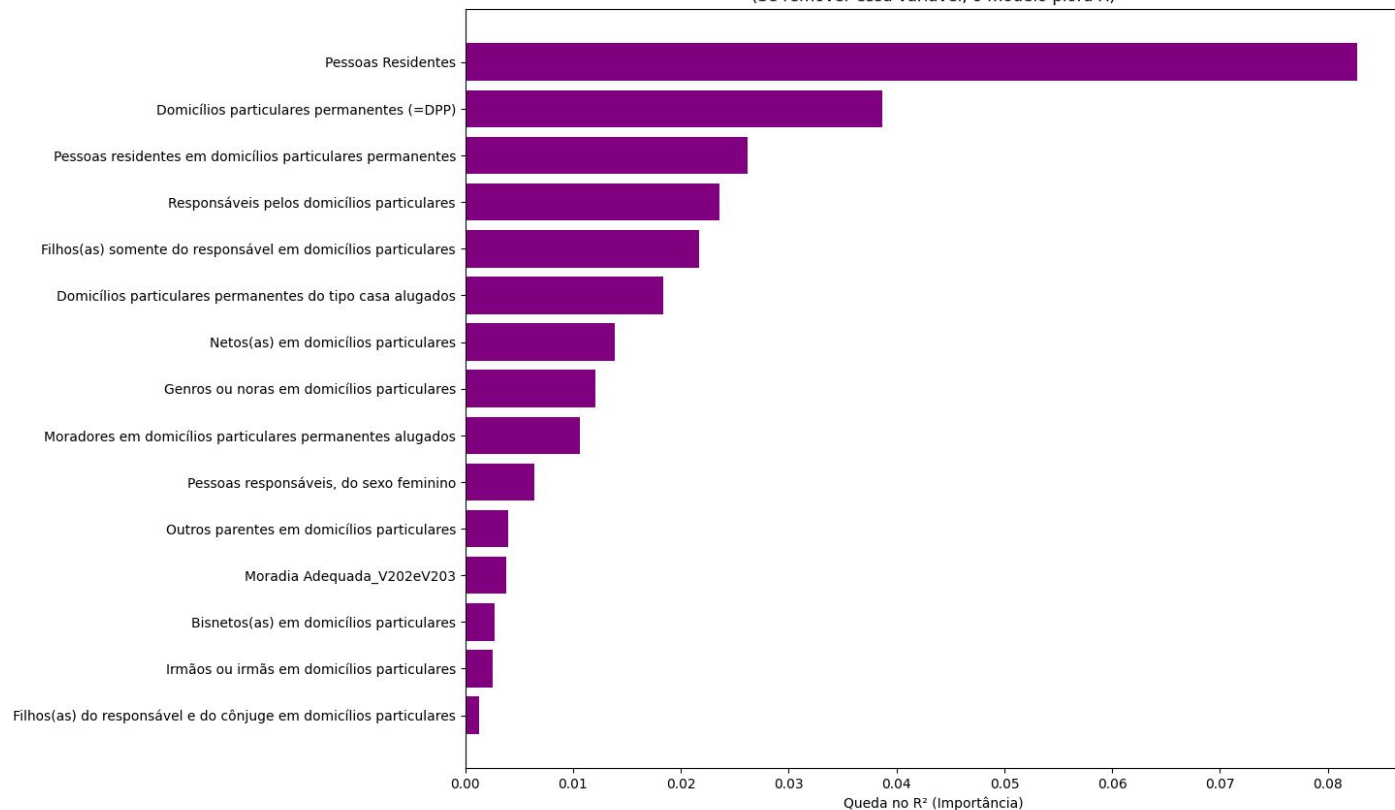
Permutation Importance - ONUS\_EXCESSIVO  
(Se remover essa variável, o modelo piora X)



Permutation Importance - ADENSAMENTO  
(Se remover essa variável, o modelo piora X)

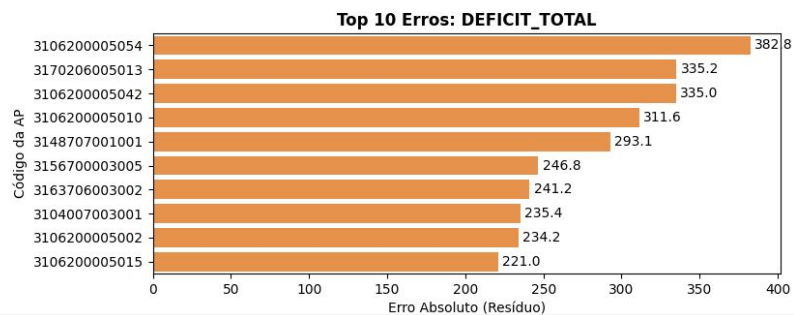
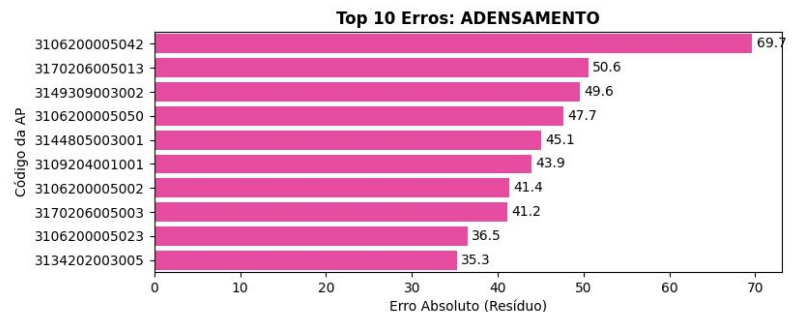
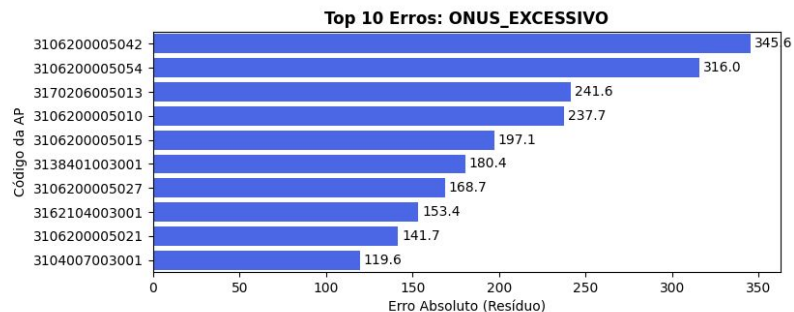
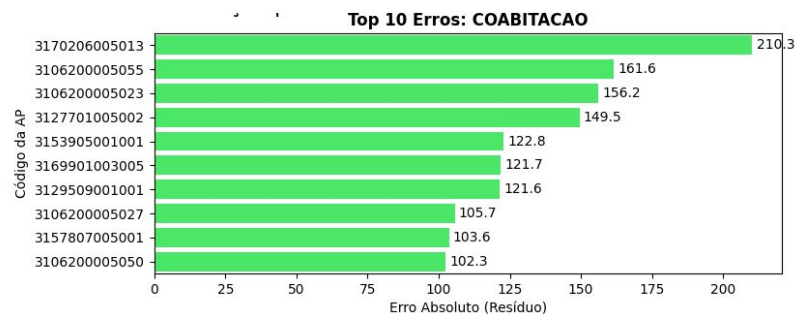
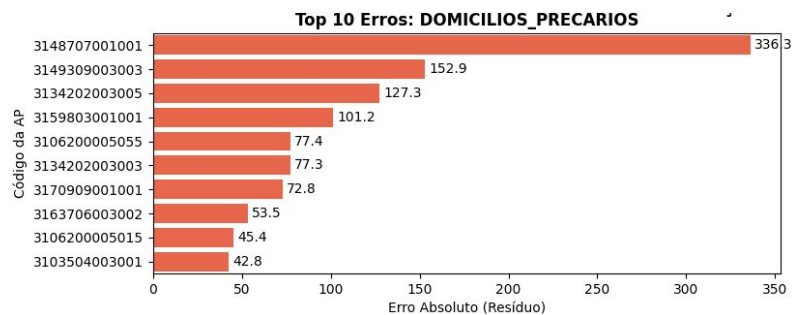


Permutation Importance - DEFICIT\_TOTAL  
(Se remover essa variável, o modelo piora X)



**5.** Mapeamento espacial dos erros:  
identificação das AP's que possuem  
maior erro e possibilidade de análise  
de erros por padrões geográficos;

---



## 6. Sobre desagregação e próximos passos

---

# Sobre desagregação e próximos passos

## 1. Erro vs Esparsidade (Porcentagem de Zeros)

Verificar se AP com muitas variáveis esparsas produz erros maiores e se há relação entre sparsity e dificuldade preditiva. Output: gráfico e regressão erro × porcentagem de zeros

### 1.1 Validação estratificada por região

## 1. Caracterização da Saída “Domicílios Precários”

Entender por que esse alvo tem desempenho baixo (ex.: alta assimetria, zeros excessivos, cauda longa). Inclui: histograma, kurtosis, skewness, dispersão

### 2.1. Encontrar modelos de predição somente para essa saída (métodos de gradiente provavelmente)

## 1. Base Conceitual para Desagregação (Setores Censitários)

Com o banco de setores censitários preparar metodologia de downscaling atuando com ambos os bancos e relacionando o valor predito por cód. ponderação em relação a cada setor censitário a ele associado

### 3.1 Definir metodologia estatística de desagregação (provavelmente a FJP já tem alguma).