

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Pedro Gabriel Lima de Andrade

**Desempenho de modelos de gradient boosting para predição de inflação
mensal**

Belo Horizonte
2021

Pedro Gabriel Lima de Andrade

**Desempenho de modelos de gradient boosting para predição de inflação
mensal**

Trabalho de Conclusão de Curso apresentado ao
Curso de Especialização em Ciência de Dados e
Big Data como requisito parcial à obtenção do
título de especialista.

Belo Horizonte

2021

SUMÁRIO

| | |
|--|-----------|
| 1. Introdução | 5 |
| 1.1. Contextualização | 5 |
| 1.2. O problema proposto | 7 |
| 1.3. Objetivos | 8 |
| 1.4 Ferramentas e tecnologia | 8 |
| 2. Coleta de Dados | 9 |
| 2.1. Dados do SGS | 9 |
| 2.2. Dados do Sistema de Expectativas de Mercado | 11 |
| 3. Processamento/Tratamento de Dados | 12 |
| 4. Análise dos dados | 14 |
| 4.1 Índices de preço | 14 |
| 4.2 Expectativas inflacionárias | 18 |
| 5. Modelagem | 21 |
| 5.1 Amostragem | 21 |
| 5.2 Gradient boosting | 22 |
| 5.3 Métrica de erro | 23 |
| 5.4 Otimização bayesiana para hiperparâmetros | 23 |
| 5.5 Arquitetura de modelagem | 25 |
| 5.6 Curvas de convergência | 26 |
| 6. Apresentação dos Resultados | 30 |
| 6.1 Descrição dos resultados | 30 |
| 6.1.1 Predição de um mês | 31 |
| 6.1.2 Predição de dois meses | 33 |
| 6.1.3 Predição de três meses | 34 |
| 6.1.4 Predição de quatro meses | 35 |
| 6.2. Importância das variáveis | 37 |
| 8. Links | 43 |

1. Introdução

1.1. Contextualização

No ano de 2013, a *Valor Econômico* publicou uma matéria, escrita por RIBEIRO (2013), intitulada “O Samba dos juros”, em alusão ao novo modelo de projeção econômica do Banco Central do Brasil (BACEN). Este modelo utiliza técnicas da fronteira de macroeconomia para prever e descrever aspectos importantes do estado da economia brasileira, como inflação, mercado de trabalho, setor externo e entre outros.

REPORTAGEM DE CAPA

O Departamento de Pesquisa do Banco Central emprega um modelo econômico tropical e um batalhão de Ph.Ds para fazer suas estimativas econômicas. Por [Alex Ribeiro](#), de Brasília

O Samba dos juros

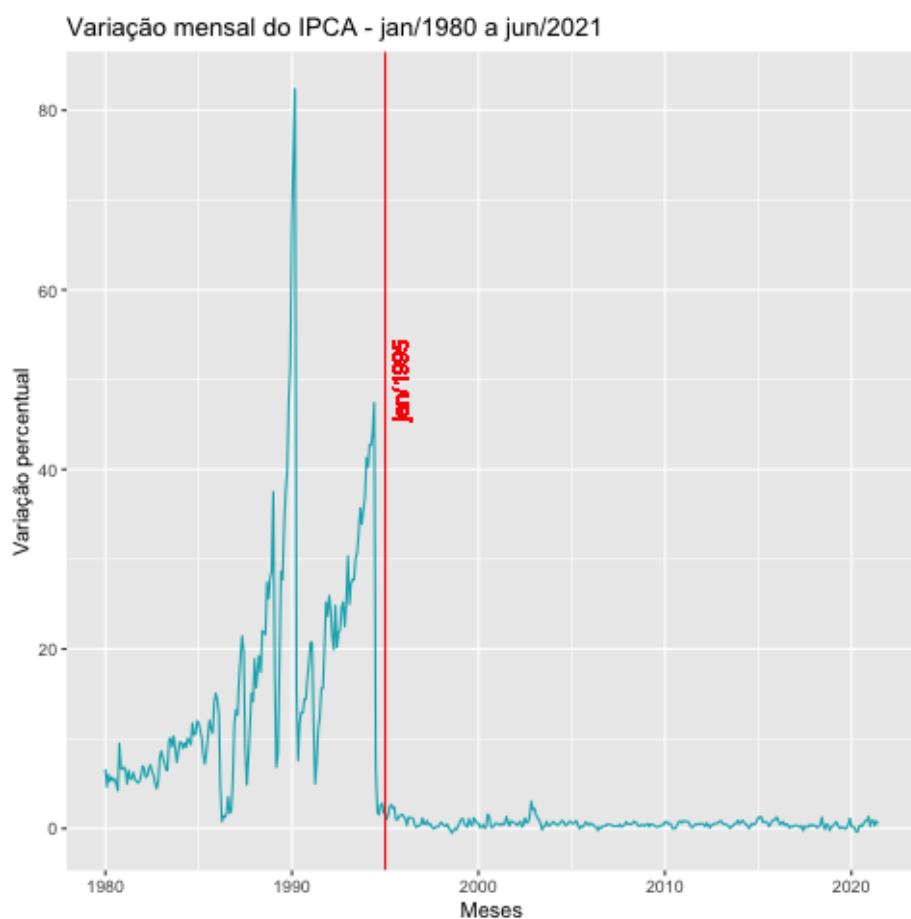
Com esse e outros modelos, o Banco Central conseguiu tornar a condução de política monetária mais assertiva, auxiliando na predição da inflação a cada ciclo de decisão do Comitê de Política Monetária. Além disso, esses modelos geram informações gerenciais para outras decisões, como comportamento do mercado de trabalho, comércio exterior e entre outros.

Não apenas para o Banco Central, mas para milhares de agentes econômicos, a previsão em relação à inflação é importante para o planejamento orçamentário, tanto para o setor privado como também para esfera governamental. O setor público, por exemplo, utiliza expectativas inflacionárias como insumo para gestão de execução orçamentária, e também na confecção da peça do orçamento anual, incluindo as previsões em um documento chamado Lei de Diretrizes Orçamentárias.

Ademais, agentes privados, como empresas, ajustam estoques ou políticas contábeis com base em expectativas inflacionárias, e famílias conseguem se planejar

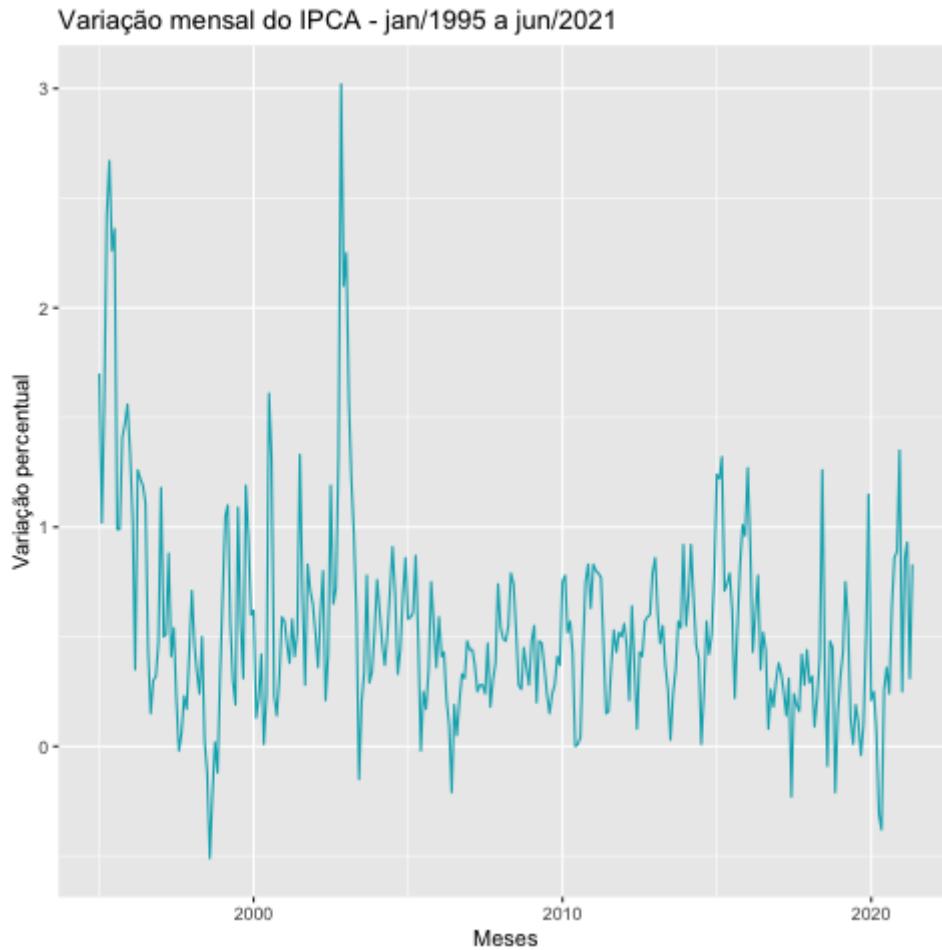
melhor tendo essas expectativas ancoradas. Portanto, subentende-se que a previsão da inflação mensal é um componente que agrega positivamente as decisões de toda a população.

Como podemos ver na imagem abaixo, é um problema visível que impactou de maneira significativa a sociedade brasileira, que atualmente necessita de manter expectativas inflacionárias ancoradas para que os problemas não se repitam. Portanto, com ferramentas de ciência de dados disponibilizadas, abre-se uma nova fronteira de aplicação dessas técnicas no contexto de previsão da inflação.



1.2. O problema proposto

O problema deste trabalho envolve a predição do valor de variação mensal do IPCA (Índice Nacional de Preços ao Consumidor Amplo), que é o principal índice que identifica variação de preços de maneira generalizada na economia brasileira.



Para prever esse cenário, será utilizado algoritmos de *gradient boosting*, que são notáveis por possuírem boa performance em problemas envolvendo dados tabulares. Desses algoritmos, escolheu-se os três mais importantes e que possuem melhor performance: XGBoost, CatBoost e LightGBM. Com isso, será observado o desempenho desses modelos, comparando-os com as expectativas de mercado sobre a variação do IPCA.

1.3. Objetivos

O objetivo central deste trabalho é identificar a performance dos melhores modelos de *gradient boosting* na previsão da inflação mensal do Brasil.

Os objetivos secundários são:

- Criar um *scrapping* para coletar os dados necessários a partir das APIs do Banco Central;
- Demonstrar os métodos de *feature engineering* adequados para os modelos;
- Criar os modelos de *gradient boosting* ajustando hiperparâmetros através da otimização bayesiana;
- Obter os resultados com a métrica do erro quadrático médio;
- Analisar os resultados extraíndo as *features* mais importantes.

1.4 Ferramentas e tecnologia

As ferramentas e técnicas utilizadas para esse trabalho serão:

- Linguagem python para coletagem e modelagem;
- Linguagem R para visualização dos dados;
- SQLite para armazenar os dados coletados no scrapping;
- Otimização bayesiana para otimizar os hiperparâmetros dos modelos, com a biblioteca Scikit-Optimize;
- Catboost, XGBoost e LightGBM para modelos de gradient boosting;
- Biblioteca SHAP para extrair as features mais importantes;

2. Coleta de Dados

Para previsão de inflação, há uma literatura extensa sobre quais variáveis são preditoras de variação de preços em uma economia. Os dados utilizados nesse trabalho foram escolhidos com base no trabalho de TERRA e PASSADOR (2012), que aplicou redes neurais na previsão de inflação para pequenos negócios. Neste trabalho, os autores optaram pela abordagem de NAKANO e BRESSER-PEREIRA (1984) sobre os fatores que interferem na inflação, que basicamente são: taxa de juros básica da economia, taxa de câmbio, crescimento dos salários, resultado primário do setor público, base monetária restrita, crescimento do PIB e inflação anterior.

Para se buscar estes dados, coletou-se de dois sistemas de armazenamento e recuperação de dados financeiros e econômicos: o Sistema Gerenciador de Séries Temporais (SGS) e o Sistema de Expectativas de Mercado, ambos mantidos pelo Banco Central do Brasil. No SGS, é possível recuperar dados de vários temas atinentes à ciência econômica, como dados de emprego, preços, economia internacional, mercado monetário e entre outros. O Sistema de Expectativas de Mercado, por sua vez, armazena dados de expectativas de agentes de mercado sobre algumas variáveis da economia brasileira, como os principais índices de preços e taxa de juros.

2.1. Dados do SGS

A coleta desses dados dividiu-se em bases de dados de frequência diária e mensal, com predominância desta última. Os dados se referem à majoritariamente índices de preços, como o Índice Nacional de Preços ao Consumidor (INPC) e Índice Geral de Preços ao Mercado (IGP-M), que possuem metodologias diferentes, para buscar variação de preços em setores diversos. Estes indicadores possuem desagregações, que podem ser vistas na lista de todas as variáveis usadas na coleta do SGS, que consta no apêndice deste trabalho.

Além disso, buscou-se dados de nível de atividade econômica, como o IBC-BR e o Índice de Condições Econômicas Atuais, ambos indicadores sintetizados pelo Banco Central. Para dados referentes a finanças públicas, ou mais especificamente superávit primário, coletou-se a série do NFSP (Necessidade de Financiamento do Setor Público) para superávit primário, que é um indicador de variação do estoque da dívida do setor público brasileiro. Além disso, buscou-se dados de base monetária restrita e também do custo da cesta básica por capitais.

Os nomes das variáveis e o respectivo código no SGS constam no dicionário abaixo. Este dicionário alimenta a função que coleta esses dados via API do Banco Central.

```

num_series_bacen_m = {"index_exp_futuras": 4395, "index_confianca": 4393,
"index_cond_econ_atuais": 4394, "ibc_br": 24363, "nfsp_rp": 4649, "ipca_comerc": 4447, "ipca_nao_comerc": 4448,
"ipca_itens_livres": 11428, "ipca_servicos": 10844, "ipca_duraveis": 10843, "ipca_bens_semidur": 10842,
"ipca_nao_duraveis": 10841, "inpc_index_dif": 21379, "inpc_nucleo_suav": 4466, "igpm": 189, "igpm_di": 190,
"ipc_br": 191, "incc": 192, "ipa": 225, "ipc_nucleo": 4467, "igp10": 7447, "igpm_1decendio": 7448, "igpm_2decendio": 7449,
"ipam": 7450, "ipam_1decendio": 7451, "ipam_2decendio": 7452, "ipcm": 7453, "ipcm_1decendio": 7454, "ipcm_2decendio": 7455,
"incc": 7456, "incc_1decendio": 7457, "incc_2decendio": 7458, "ipaog_prod_indus": 7459, "ipaog_prod_agro": 7460,
"inpc": 188, "ipc_fipe": 193, "ipc_fipe_2quadrisemana": 272, "ipca": 433, "ipca_alim_beb": 1635, "ipca_habit": 1636,
"ipca_art_habit": 1637, "ipca Vestuario": 1638, "ipca_transportes": 1639, "ipca_comunicacao": 1640, "ipca_saude": 1641,
"ipca_desp_pes": 1642, "ipca_educacao": 1643, "inpc_alim_beb": 1644, "inpc_habit": 1645, "inpc_art_habit": 1646,
"inpc_vestuario": 1647, "inpc_transporte": 1648, "inpc_comunicacao": 1649, "inpc_saude": 1650, "inpc_desp_pes": 1651,
"inpc_educacao": 1652, "ipca_monitorados": 4449, "ipc_fipe_1quadrisemana": 7463, "ipc_fipe_3quadrisemana": 7464,
"ipc_fipe_aliment": 7465, "ipc_fipe_indust": 7467, "ipc_fipe_innatura": 7468, "ipc_fipe_habit": 7469,
"ipc_fipe_transp": 7470, "ipc_fipe_desp_pes": 7471, "ipc_fipe_vest": 7472, "ipc_fipe_saude": 7473,
"ipc_fipe_educacao": 7474, "ipc_fipe_comerc": 7475, "ipc_fipe_nao_comerc": 7476, "ipc_fipe_monit": 7477,
"ipca15": 7478, "ipca_e": 10764, "ipca_12meses": 13522, "ipca_industriais": 27863, "ipca_alim_dom": 27864,
"custo_cesta_aracaju": 7479, "custo_cesta_belem": 7480, "custo_cesta_bh": 7481, "custo_cesta_brasilia": 7482,
"custo_cesta_curitiba": 7483, "custo_cesta_floripa": 7484, "custo_cesta_fortaleza": 7485, "custo_cesta_goiania": 7486,
"custo_cesta_jp": 7487, "custo_cesta_natal": 7488, "custo_cesta_poa": 7489, "custo_cesta_recife": 7490,
"custo_cesta_rj": 7491, "custo_cesta_salvador": 7492, "custo_cesta_sp": 7493, "custo_cesta_vitoria": 7494,
"pib_acum_12meses": 4192, "pib_mensal": 4380, "pib_acum_ult12meses": 4382}

^ num_series_bacen_d = {"selic_diaria": 11, "selic_acum": 1178, "meta_selic": 432}

```

Estes dois dicionários alimentam a função abaixo, que busca os dados nas APIs do Banco Central.

```

def carga_dados_historicos(mes, ano, n_serie_m, n_serie_d):
    database = pd.DataFrame(index=pd.PeriodIndex(pd.date_range(start='1995-1-1', end=f'{ano}-{mes}-01', freq='M'), freq="M"))

    for i in tqdm(n_serie_m.keys(), desc="Carga de dados mensais"):
        n = num_series_bacen_m[i]
        data = pd.read_json(f"https://api.bcb.gov.br/dados/serie/bcdata.sgs.{n}/dados?formato=json&dataInicial=01/01/1995&dataFinal=01/{mes}/{ano}")
        data = data.set_index(pd.PeriodIndex(pd.to_datetime(data["data"], format="%d/%m/%Y"), freq="M")).drop(["data"], axis=1)
        database[i] = data["valor"]

    for i in tqdm(n_serie_d.keys(), desc="Carga de dados diários"):
        n = num_series_bacen_d[i]
        data = pd.read_json(f"https://api.bcb.gov.br/dados/serie/bcdata.sgs.{n}/dados?formato=json&dataInicial=01/01/1995&dataFinal=11/{mes}/{ano}")
        data["data"] = pd.to_datetime(data["data"], format='%d/%m/%Y')
        data = data.set_index("data").resample('M').mean().reset_index()
        data = data.set_index(pd.PeriodIndex(data["data"], freq="M")).drop(["data"], axis=1)
        database[i] = data["valor"]

    database = database.reset_index()
    database["index"] = database["index"].astype(str)
    database = database.rename(columns={"index": "date"})

    conn = sqlite3.connect("data/app_db.db")
    database.to_sql(name="dados", con=conn, if_exists='replace')

    return database

```

2.2. Dados do Sistema de Expectativas de Mercado

No que se refere aos dados de expectativas de mercado, colheu-se a média, mediana e desvio padrão das expectativas para quatro índices: IGP-M, IGP-DI, INPC e IPCA. O horizonte de tempo de previsão foi de cinco meses, ou seja, a partir de um determinado mês, coletou-se os dados das expectativas para os 5 meses seguintes. Estes dados, no entanto, só possuem informações a partir do ano 2000, quando se criou o sistema. Com isso, criou-se uma janela entre 1995 até 2000 que teria que ser resolvido via técnica de feature engineering. O script de coleta dos dados de expectativas consta abaixo.

```

def carga_dados_expectativas(mes, ano):
    conn = sqlite3.connect("../data/app_db.db")
    database = pd.read_sql("select * from dados", conn)
    database = database.set_index(["date"])
    dics = {'IGP-DI': 'igp_di', 'IGP-M': 'igp_m', 'INPC': 'inpc', 'IPCA': 'ipca'}
    periodo = pd.date_range(start='2000-01-01', end=f'{ano}-{mes}-01', freq='M')

    for d in dics.values():
        for l in range(1, 6):
            database[d + '_mean_' + str(l)] = None
            database[d + '_median_' + str(l)] = None
            database[d + '_dp_' + str(l)] = None

    for i in dics.keys():

        for o in tqdm(range(periodo.shape[0]), desc=i):

            try:
                d = periodo[o].strftime('%Y-%m-%d')

                for n in range(1, 6):
                    ds = pd.date_range(start = d, periods=6, freq='M')[n].strftime('%Y-%m')
                    y = ds.split('-')[0]
                    m = ds.split("-")[1]

                    r = requests.get(f"https://olinda.bcb.gov.br/olinda/servico/Expectativas/versao/v1/odata/ExpectativaMercadoMensais?$top=1" +
                                    "&$filter=Indicador%20eq%20'{i}'%20and%20DataReferencia%20eq%20'{m}%2F{y}'%20and%20data%20le%20'{d}'&$orderby=Data%20desc&$" +
                                    "format=json&$select=Indicador,Data,DataReferencia,Media,Mediana,DesvioPadrao,CoefficienteVariacao,Minimo,Maximo")

                    x = d.split('-')[0] + '-' + d.split('-')[1]
                    data = pd.DataFrame(r.json())["value"]
                    database[dics[i] + '_mean_' + str(n)].loc[x] = data.loc[0]['Media']
                    database[dics[i] + '_median_' + str(n)].loc[x] = data.loc[0]['Mediana']
                    database[dics[i] + '_dp_' + str(n)].loc[x] = data.loc[0]['DesvioPadrao']

            except:
                pass

    database = database.drop("index", 1)
    database = database.reset_index()
    database["date"] = database["date"].astype(str)
    conn = sqlite3.connect("../data/app_db.db")
    database.to_sql(name="dados", con = conn, if_exists='replace', index=False)

    return database

```

3. Processamento/Tratamento de Dados

O processamento dos dados coletados foi uma tarefa relativamente simples, devido à baixa dimensionalidade dos dados. Foram feitos testes com algumas formas de engenharia de variáveis voltadas à previsão de séries temporais, como extração da média móvel e diferenças, mas se observou a perda de informações e os modelos tinham performance pior se comparada à estrutura mais simples dos dados.

Optou-se, portanto, por duas técnicas de engenharia de variáveis: a inclusão de valores faltantes e a inclusão de lags da série temporal, tanto no *target* quanto como *feature* do modelo. Para valores faltantes, buscou-se apenas incluir valores exorbitantes em espaços nos quais as séries não possuíam valores, como séries de expectativas e algumas das séries históricas do Banco Central.

Para criar lags das séries, ou seja, valores diferentes do mês corrente, construiu-se variáveis objetivos (*target*) e também variáveis como *features* do modelo. Para se construir o *target*, construiu-se 4 variáveis, referentes a quatro meses à frente os valores do mês corrente da série do IPCA, que teve o nome de "y" + o número do lag da série. Por exemplo, para se prever dados do IPCA de três meses à frente, constrói-se, com a função "shift" da biblioteca pandas, o lag da série de três meses à frente do registro dos dados, denominada "y3".

Para se criar o lag dos dados como *feature*, optou-se por incluir o valor do mês anterior de três índices: IPCA, INPC e IGP-M, os mais importantes das séries de preços do país. Tentou-se outras formas de incluir com outras variáveis, mas se perdia em performance dos modelos. O script para criar os datasets, que foram os mesmos para os três algoritmos, consta na imagem abaixo, com a função "criar_dataset".

```
def criar_dataset(lag):
    conn = sqlite3.connect("../data/app_db.db")
    dados = pd.read_sql("select * from dados", conn)

    dados["y1"] = dados["ipca"].shift(-1)
    dados["y2"] = dados["ipca"].shift(-2)
    dados["y3"] = dados["ipca"].shift(-3)
    dados["y4"] = dados["ipca"].shift(-4)

    l_cols = ["ipca", "inpc", "igpm"]

    for i in l_cols:
        dados[i + "shift"] = dados[i].shift(1)

    for i in dados.drop(["y1", "y2", "y3", "y4"], axis=1).columns:
        dados[i] = dados[i].fillna(-999999999999)

    dados = dados.set_index("date")
    dados = dados.dropna(subset=[f"y{lag}"])

    return dados
```

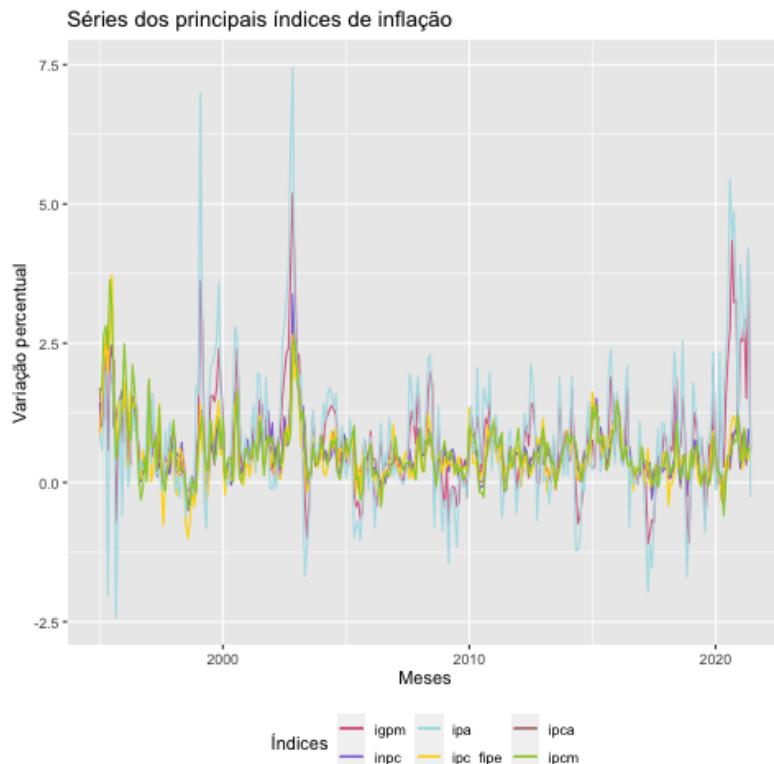
4. Análise dos dados

Nesta seção, serão elucidados alguns indicadores que compõem a base de dados constituída na seção anterior. O conjunto de dados que serviu de base para os algoritmos deste trabalho tem uma dimensão de 317 linhas e 194 colunas, com *features* tanto relacionadas aos dados históricos quanto de expectativas de mercado.

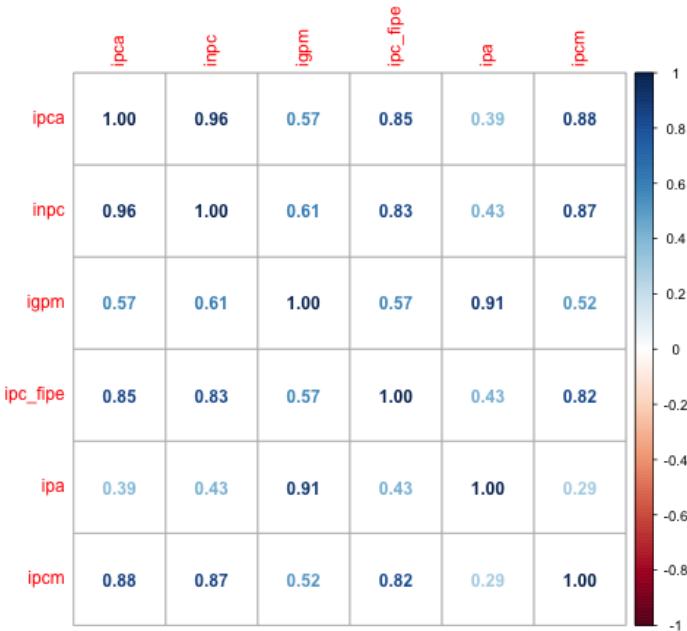
4.1 Índices de preço

Os índices de preços e suas desagregações, como se espera em um problema de predição de inflação, são variáveis mais presentes no problema aqui abordado. Há variáveis como os principais índices de preços - INPC, IPCA, IGP-M e entre outros -, índices de custos, e outras variáveis relacionadas a este tema. Será elucidado aqui dados de desagregações dessas variáveis, como componentes de alimentação, energia e outros dos índices de preços.

No gráfico abaixo, por exemplo, é possível observar o comportamento destes índices ao longo dos anos, mensurado pela taxa de variação. Com exceção de momentos específicos, como a crise recente, os índices tendem a ter comportamentos parecidos, mas que podem guardar informações desagregadas poderosas para predição.

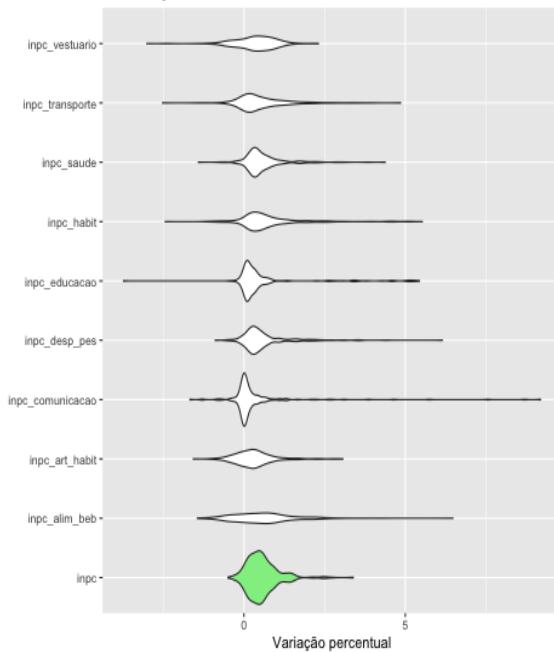


No que se refere a relação entre essas variáveis, na matriz de correlação abaixo é possível identificar a relação da variação entre elas ao longo do tempo, em nível de intensidade. Índices como IPCA e INPC possuem correlação forte, enquanto IPA (Índice de Preços do Atacado) e IPCM (Índice de Preços ao Consumidor - Mercado) possuem fraca correlação, que ambas as situações podem auxiliar os algoritmos a tomarem as decisões sobre inflação mensal.

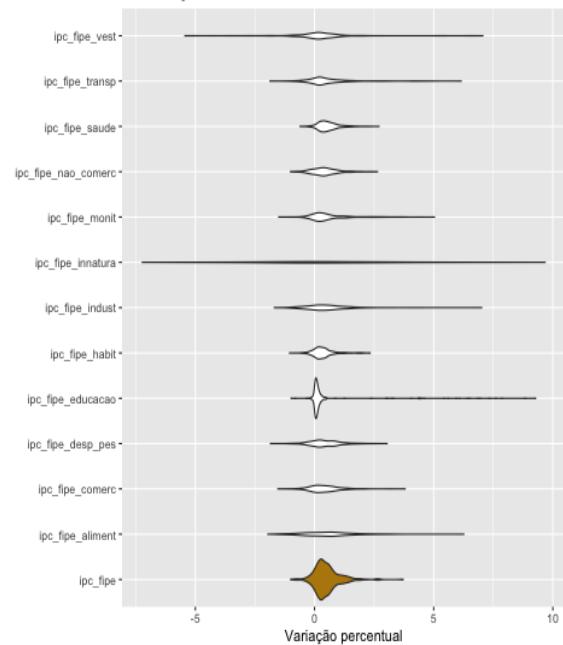


Com relação à desagregação dos índices de inflação, a imagem abaixo consta gráficos de distribuição da variação de cada índice que compõem três índices fundamentais nesse estudo: INPC, IPCA e o IPC-FIPE. É possível observar que nos próprios índices há divergência de comportamento entre os indicadores que o compõem, com alguns se ajustando mais à média e outros mais voláteis, a depender do setor no qual se encontram.

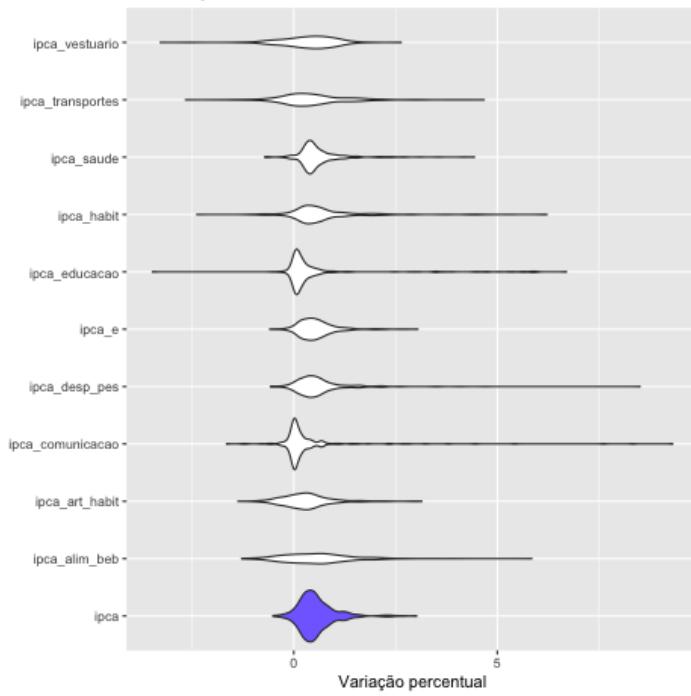
Distribuição das variáveis do INPC



Distribuição das variáveis do IPC-FIPE

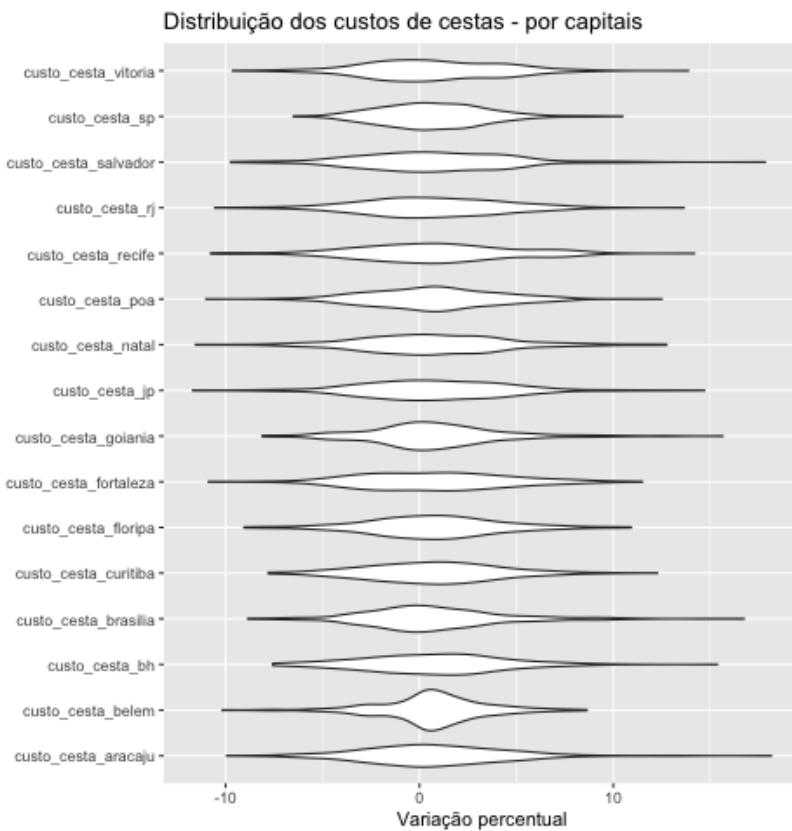


Distribuição das variáveis do IPCA



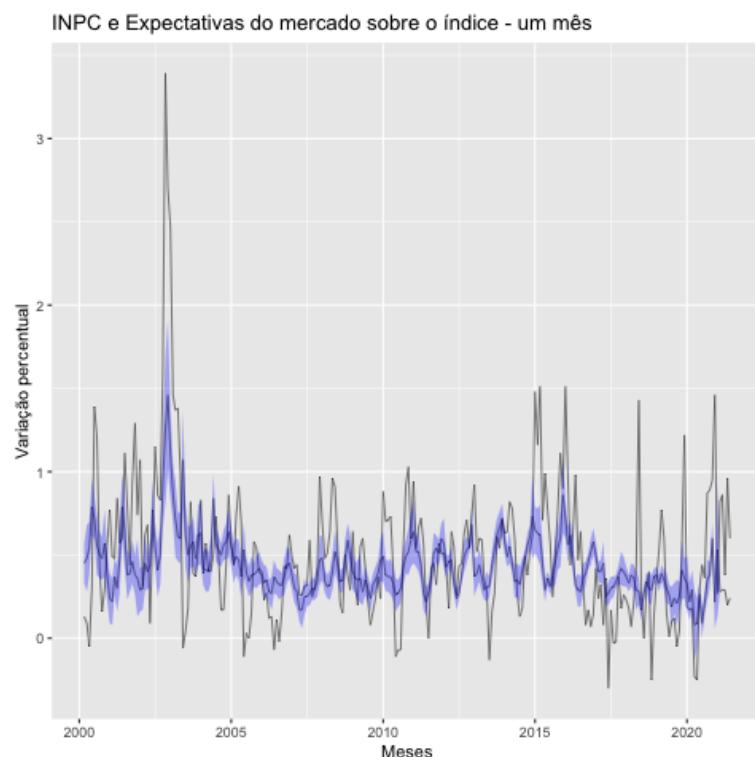
Ainda sobre variação de índices, há um índice interessante que se trata do custo da cesta básica por capital dos estados federativos. Com esse índice é possível acompanhar de maneira desagregada as mudanças no valor de suprimentos básicos em várias localidades do Brasil, o que permite que o algoritmo talvez antecipe mudanças inflacionárias conforme os custos vão se alterando.

No gráfico abaixo, da distribuição da variação desses custos ao longo dos meses e anos, observa-se que apesar de todos seguirem uma média de variação similar, possuem desvios diferentes e concentração da variação também diferenciada. Os custos da cesta de São Paulo em comparação a Salvador possuem comportamentos distintos, o que pode configurar impactos em termos de inflação mensal, por isso foi incluído como feature nos modelos.

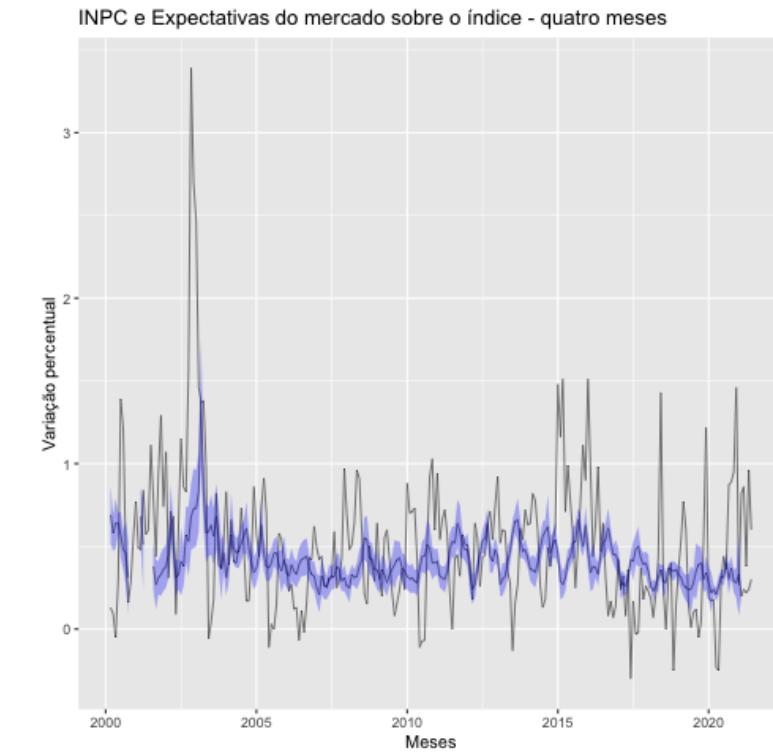


4.2 Expectativas inflacionárias

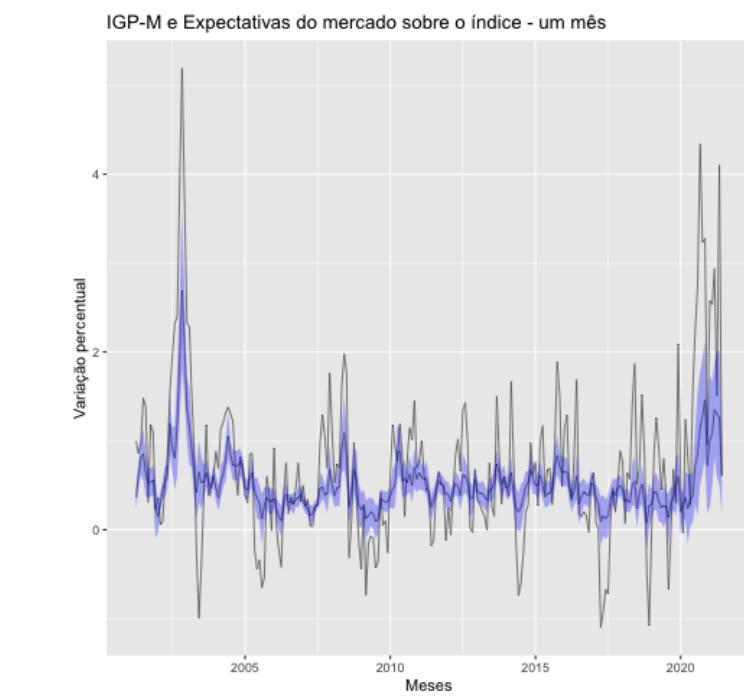
Como foi mencionado, os dados também contém as expectativas dos agentes de mercado sobre os índices de preços. No gráfico abaixo, por exemplo, consta a média da expectativa do mercado com o respectivo desvio padrão para o INPC de um mês, ou seja, o que o mercado esperava sobre esse índice um mês anterior à sua ocorrência. As linhas em preto correspondem aos dados do IPCA, enquanto as linhas azuis correspondem a média da expectativa do mercado sobre o índice, com as bandas de certezas, as quais são

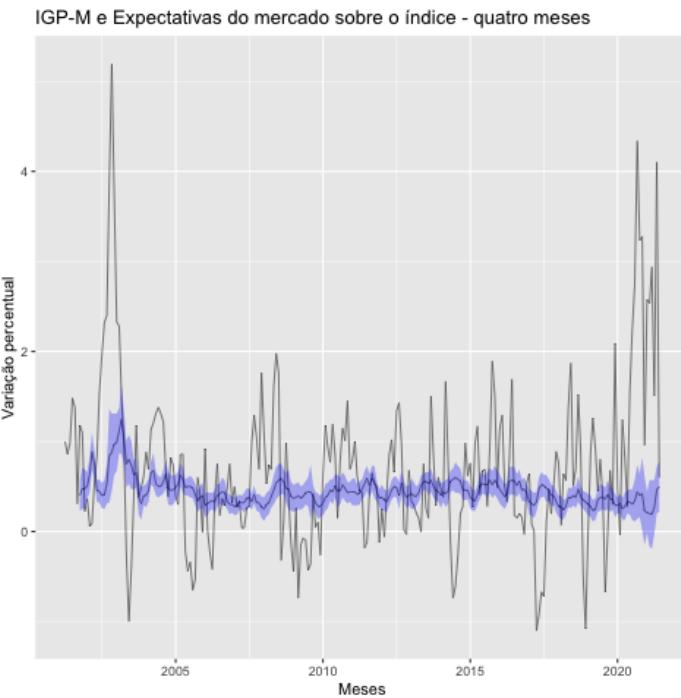


Se comparado com a previsão quatro meses, é possível observar que há diferenças no grau de certeza do mercado sobre o comportamento do índice. Devido a isso é importante ter novos modelos, como este trabalho, para que possam aprimorar esse horizonte de predição, para que torne as percepções menos incertas.



O IGP-M, por sua vez, por ser um índice mais abrangente de preços, que abarca índice ao consumidor, atacado e custos em geral, evidentemente carrega mais incertezas, justamente por agregar múltiplos setores. Isso se observa na diferença no comportamento das expectativas de mercado, que em um prazo de quatro meses é bem mais incerta quanto ao valor da variação se comparado com a previsão para apenas um mês.





As análises exploratórias desses dados não são exaustivas, evidentemente. Há muitas variáveis para alimentar os modelos, como câmbio, nível de atividade econômica e entre outros – e todas elas constam no apêndice deste trabalho. Com essa análise foi possível observar o comportamento de algumas variáveis, principalmente ao longo do tempo e com diferentes perspectivas setoriais e locais.

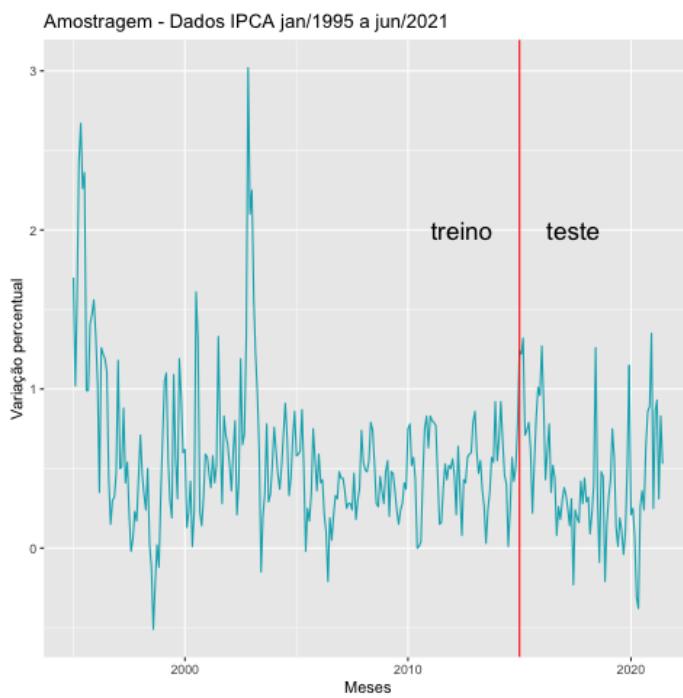
5. Modelagem

Nessa seção, será demonstrado todo o processo de modelagem feito, com amostragem, explicação breve dos modelos, métrica de erro, otimização para minimizar os erros e comportamento da curva de convergência.

5.1 Amostragem

A estratégia de amostragem deste problema possui duas restrições, o fato de existir uma dimensionalidade baixa dos dados e se tratar de predição de série temporal. Para contornar esse problema, optou-se por uma estratégia simples de amostragem, que divida os dados conforme a posição ao longo dos meses e anos e recorra a um divisão simples, que permita que os modelos possam ajustar bem as curvas às características das séries.

Em termos de composição percentual dos dados para treino e teste, a divisão ficou a seguinte maneira: 75% dos dados para treino; 25% para teste. A divisão se deu pelo critério temporal, no qual os dados de janeiro de 1995 até dezembro de 2014 corresponde aos dados para treino, e a partir de janeiro de 2015 para teste, finalizando o *dataset* em junho de 2021. Por se tratar de série temporal, observou-se que era uma amostragem adequada, pois preserva a características dos dados sucessivos ao longo do tempo. Na imagem abaixo, há uma demonstração visual de como os dados foram divididos, com a linha vermelha demarcando janeiro de 2015.



5.2 Gradient boosting

Os algoritmos de *gradient boosting* são o estado da arte no que se trata de performance para predição em problemas envolvendo dados tabulares. Quando se diz estado da arte, subtende-se que atualmente possui grande aceitação na academia e indústria para resolução deste tipo de problema. Isso se deve, principalmente, ao fato serem algoritmos que conciliam bem tempo de processamento e performance.

É um tipo de algoritmo que treina novos modelos a partir de erros de modelos anteriores, ou seja, os novos modelos fazem previsão de erros anteriores ao invés de tomar decisão independentemente o target do problema. O algoritmo dessa natureza começa fazendo previsões simples, e trabalha a correção em cima de erros residuais, obtendo uma versão mais corrigida da primeira predição mais simplificada.

Desses algoritmos, há três que se destacam atualmente: XGBoost, CatBoost e o LightGBM. Usaremos, portanto, os três para avaliar o desempenho dos modelos de gradiente boosting.

5.3 Métrica de erro

O problema tratado neste trabalho se refere a modelos de regressão, que buscam prever valores numéricos, como, por exemplo, o valor da variação percentual do índice de inflação qualquer. Para mensurar a intensidade do erro dos valores previstos pelos modelos gerados aqui se optou pela métrica mais conhecida dos problemas de regressão: o erro quadrático médio – em inglês, *mean squared error* (MSE). O cálculo dessa métrica se dá da seguinte forma:

$$MSE = \frac{1}{n} \sum \underbrace{\left(y - \hat{y} \right)^2}_{\text{The square of the difference between actual and predicted}}$$

A escolha dessa forma de calcular o erro se deve ao fato de ser uma métrica que penaliza erros maiores, que durante o processo de busca dos melhores hiperparâmetros, o algoritmo vai procurar erros menores e mais bem distribuídos na amostra. Para o problema em questão, que envolve prever inflação mensal, ter erros mais equilibrados é mais adequado para o processo de tomada de decisão.

5.4 Otimização bayesiana para hiperparâmetros

Para que os modelos se ajustem de maneira mais precisa aos dados apresentados no input, é necessário que se recorra ao ajuste dos hiperparâmetros. Segundo KAUL, GOLOVIN e KOCHANSKI (2017), os hiperparâmetros contém "os dados que controlam o processo de treinamento". Em suma, são variáveis de configuração, que podem ser aprimorados a partir da execução - por diversas vezes - de todo o processo de treinamento e coleta dos resultados do output, até chegar ao melhor resultado.

Há alguns processos que podem ser feitos para busca dos hiperparâmetros mais ajustados, como o grid search, random e a otimização bayesiana. A grid search possui o problema de ser um processo que consome muito poder computacional, em detrimento da otimização bayesiana, que torna esse processo menos exaustivo e mais assertivo (KAUL, GOLOVIN e KOCHANSKI, 2017).

A abordagem bayesiana consiste em rastrear “avaliações anteriores que usam para formar um modelo probabilístico de mapeamento de hiperparâmetros para uma probabilidade de uma pontuação na função objetivo”. Ou seja, ao invés de buscar de

maneira exaustiva os resultados, obtêm-se as melhores métricas através de análise de probabilidade com base em resultados obtidos anteriormente.

O objetivo neste trabalho, portanto, é utilizar a abordagem de otimização bayesiana, por meio da biblioteca Scikit-Optimize, também chamada de “skopt”. Além disso, utilizou-se os hiperparâmetros dos modelos e espaço de procura abaixo, para cada algoritmo respectivamente enunciado. A escolha de cada hiperparâmetro se deu com a procura por artigos que atestassem a performance do conjunto escolhido.

Para a escolha dos hiperparâmetros do XGBoost, utilizou-se o espaço utilizado por GANDHI (2019), que fez uso também da biblioteca “skopt”. Para do CatBoost, fez-se o uso do espaço de otimização de MASSARON e MARINELLI (2019), publicado em um kernel da plataforma Kaggle. Do LGBM, por sua vez, consta no artigo do (2018).

XGBoost

```
space = [(0.6, 0.7), # colsample_bylevel
         (0.6, 0.7), # colsample_bytree
         (0.01, 1), # gamma
         (0.0001, 1), # learning_rate
         (0.1, 10), # max_delta_step
         (6, 15), # max_depth
         ]
```

LGBM

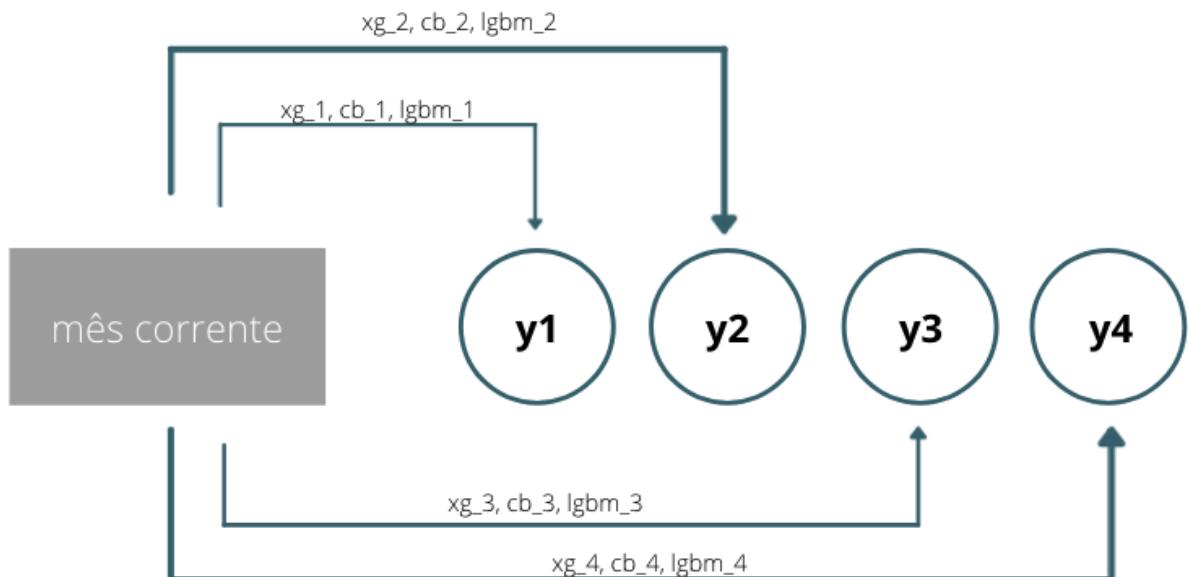
```
space = [(1e-3, 1e-1, 'log-uniform'), #learning_rate
          (2, 128), # num_leaves
          (1, 100), # min_child_samples
          (0.05, 1.0), # subsamples
          (0.1, 1.0), # colsample_bytree
          (0.1, 0.9), # feature_fraction
          (0.8, 1), # bagging_fraction
          (17, 25), # max_depth
          (0.001, 0.1), # min_split_gain
          (10, 25), # min_child_weight
          ]
```

CatBoost

```
space = [(10, 300), # iterations
          (1, 8), # depth
          (0.01, 1.0), # learning_rate
          (1e-9, 10), # random_strength
          (0.0, 1.0), # bagging_temperature
          (1, 255), # border_count
          (2, 30), # l2_leaf_reg
          ]
```

5.5 Arquitetura de modelagem

A arquitetura da modelagem focou em criar modelos separados para cada *lag* e algoritmo. Portanto, ao todo, criou-se doze modelos, que servem para prever 4 meses à frente, utilizando-se dos três algoritmos. Para isso ser feito, criou-se um script que estabelece para cada um dos 12 modelos, 300 tentativas de busca de otimização dos hiperparâmetros, que vão buscando minimizar o erro da predição com base na métrica do erro quadrático médio. A arquitetura básica de construção desses modelos consta na imagem abaixo.

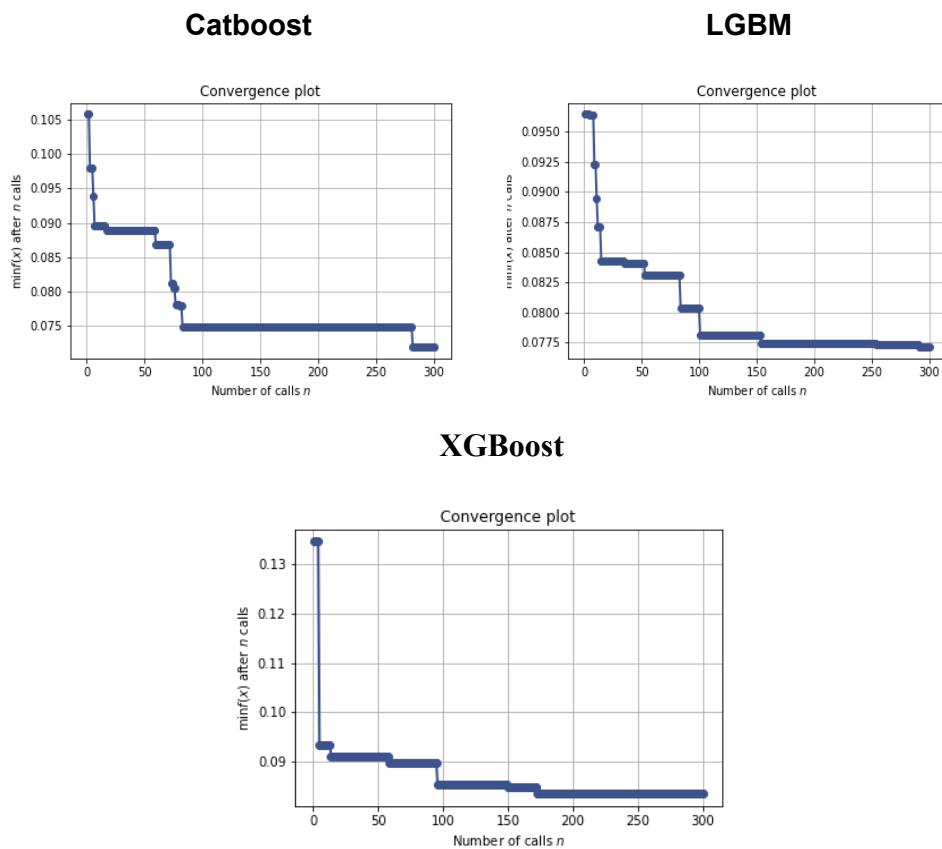


A partir do momento que se encontra a melhor combinação dos hiperparâmetros para cada modelo, este é salvo em formato “.pickle” na pasta “models” do diretório, e também se constrói a curva de convergência e também é salva no mesmo diretório.

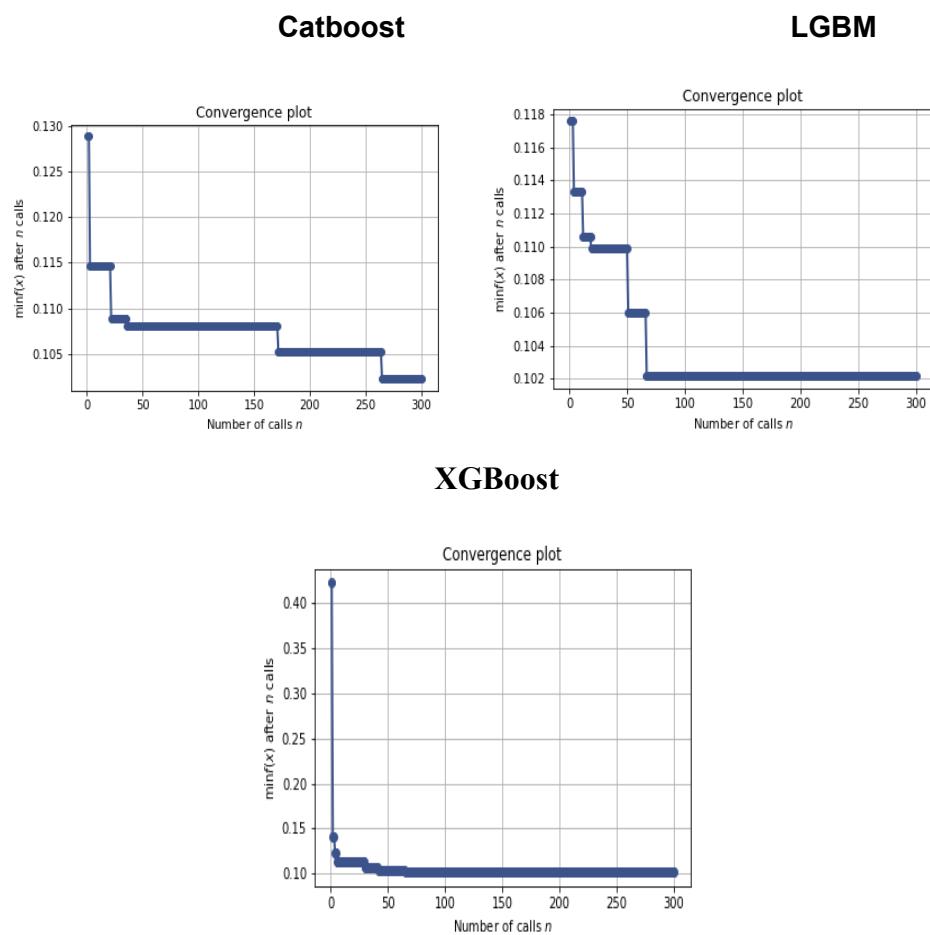
5.6 Curvas de convergência

Durante o treinamento e busca dos melhores combinações, ao longo das trezentas tentativas de combinações de hiperparâmetros, a própria biblioteca do *scikit-optimize* permite que se crie uma curva convergência para os erros mínimos ao longo do treinamento. Isso quer dizer que é possível de se acompanhar o comportamento dos algoritmos nesse treinamento e observar quando se alcança os mínimos, para que mais a frente se possa racionalizar o processo de otimização.

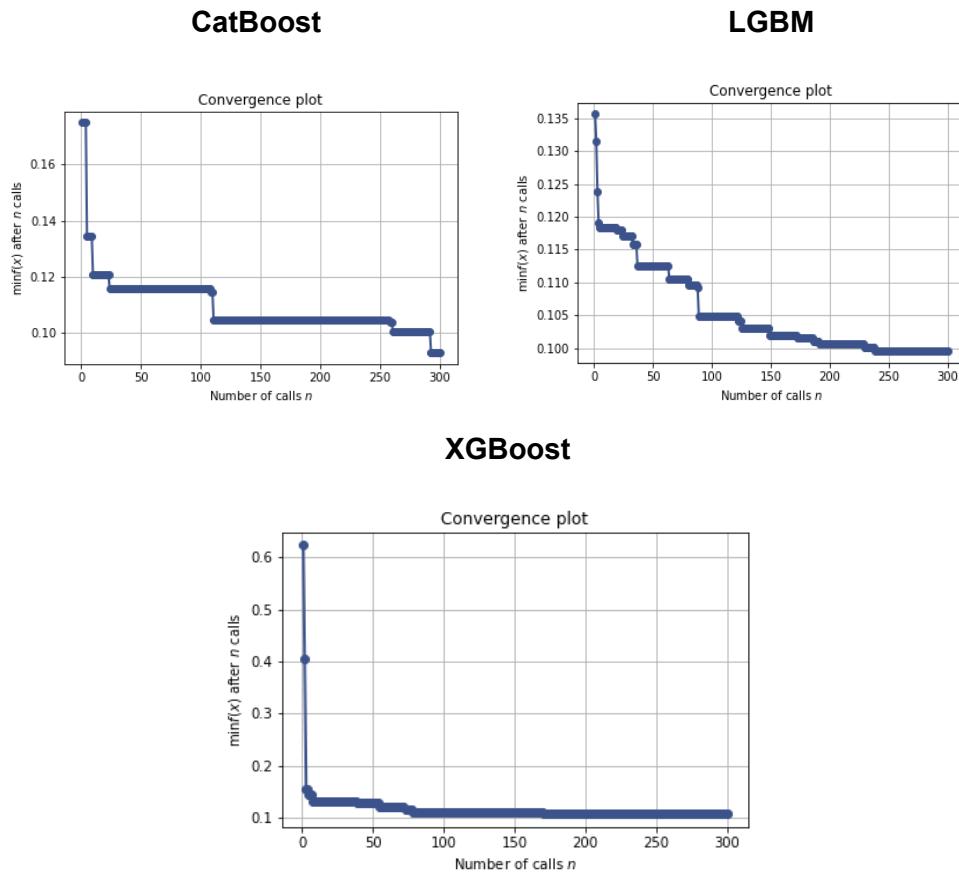
As curvas de convergência para o primeiro mês de predição são as correspondentes nos gráficos abaixo. Observa-se que o modelo de CatBoost estagnou no mesmo erro mínimo por mais de 150 tentativas, e já no final da procura conseguiu um erro ainda menor. O modelo com LGBM, por sua vez, possui uma convergência ao mínimo mais gradativa, enquanto o modelo do XGBoost teve um começo com erro alto, mas se ajustou e conseguiu uma boa performance de maneira também gradativa ao longo das tentativas.



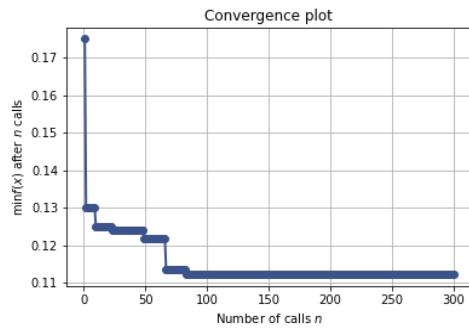
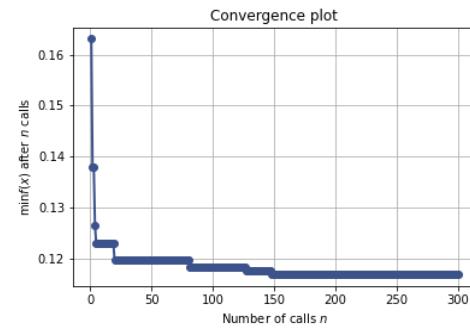
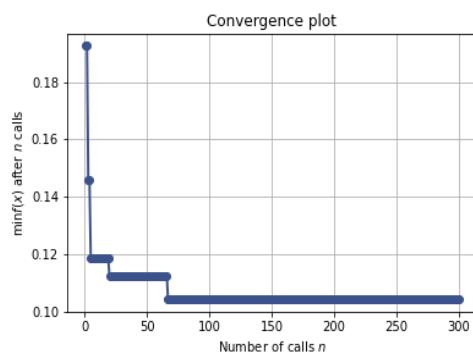
Em se tratando de modelagem para dois meses, o processo do CatBoost foi bem simétrico e gradativo em toda a série, em comparação ao LGBM que se estagnou por volta da tentativa sessenta e se manteve até o final. O modelo XGBoost, por sua vez, assim como aconteceu no modelo anterior, também teve uma rápida melhora, nas primeiras tentativas, e se estagnou no menor erro quadrático médio.



Os modelos gerados para CatBoost para três meses conseguiu um menor erro nas últimas tentativas, enquanto o LGBM teve uma convergência bastante simétrica ao longo a curva. E, como se pode notar, o XGBoost, assim como nas outras, também teve uma rápida melhor e depois se estagnou.



Em se tratando das previsões de quatro meses, o comportamento os algoritmos em termos de convergência é similar. Todos eles começaram com um erro quadrático médio bem alto, e rapidamente se ajustaram a um erro razoável e mantiveram aquela performance até o final da otimização.

CatBoost**LGBM****XGBoost**

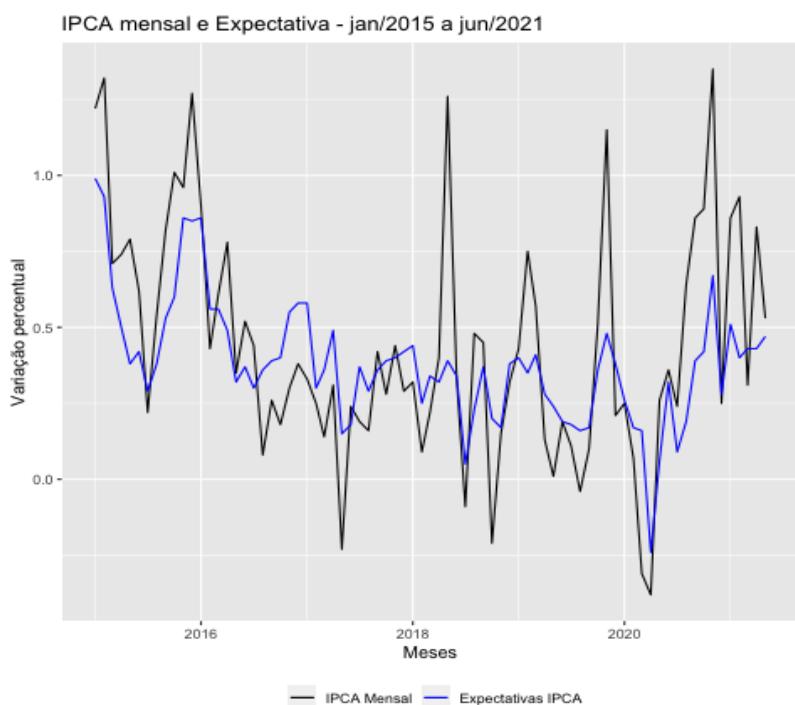
6. Apresentação dos Resultados

Nesta seção, serão apresentados dos resultados e também a análise destes resultados à luz do *feature importance* da biblioteca SHAP, que visa dar mais transparência aos resultados destes modelos.

6.1 Descrição dos resultados

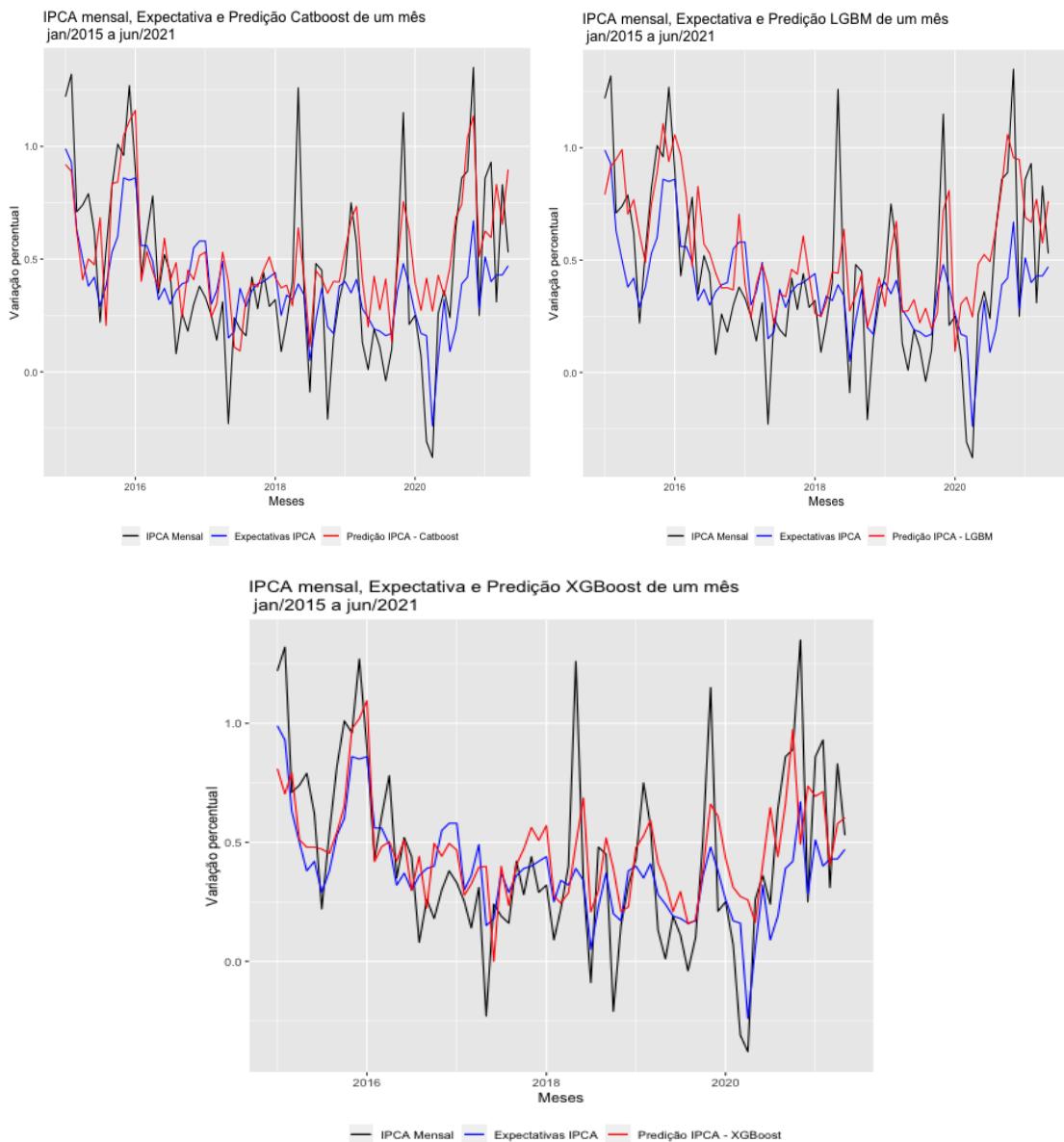
A descrição e análise dos resultados das saídas dos modelos requer um parâmetro de comparação. Para isso, optou-se por buscar dados de expectativas de mercado sobre o IPCA, para os quatro meses a frente do mês corrente e comparar com a previsão dos modelos aqui desenvolvidos. Com isso, teremos um *baseline*, uma linha que estabelece se temos modelos suficientes para serem utilizados ou basta a média do mercado sobre IPCA.

Na imagem abaixo, consta a série do IPCA e a Expectativa de um mês a frente do mês corrente. Ao longo da análise será evidenciado o nível de erro dessas expectativas do mercado e até que ponto os modelos conseguem se sobressair e ter um erro quadrático médio menor.

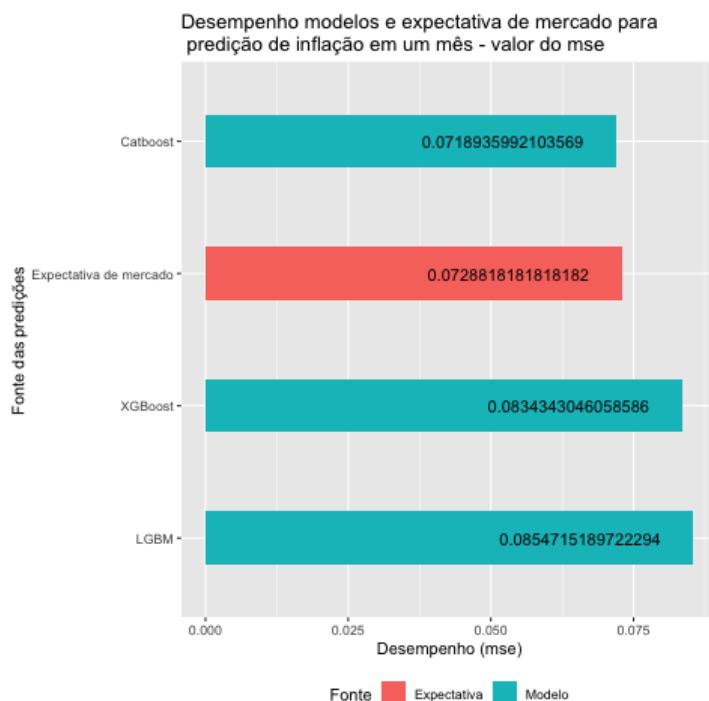


6.1.1 Predição de um mês

Observa-se nas curvas de predição abaixo o bom ajuste dos modelos construídos aqui à curva de IPCA, com destaque para o modelo de CatBoost. O grande desafio desses modelos é o ajuste a valores que fogem à média, de valores exorbitantes, que o sistema de expectativas capta bem quando se trata de valores negativos, e os modelos se ajustam mais pra valores exorbitantes positivos.

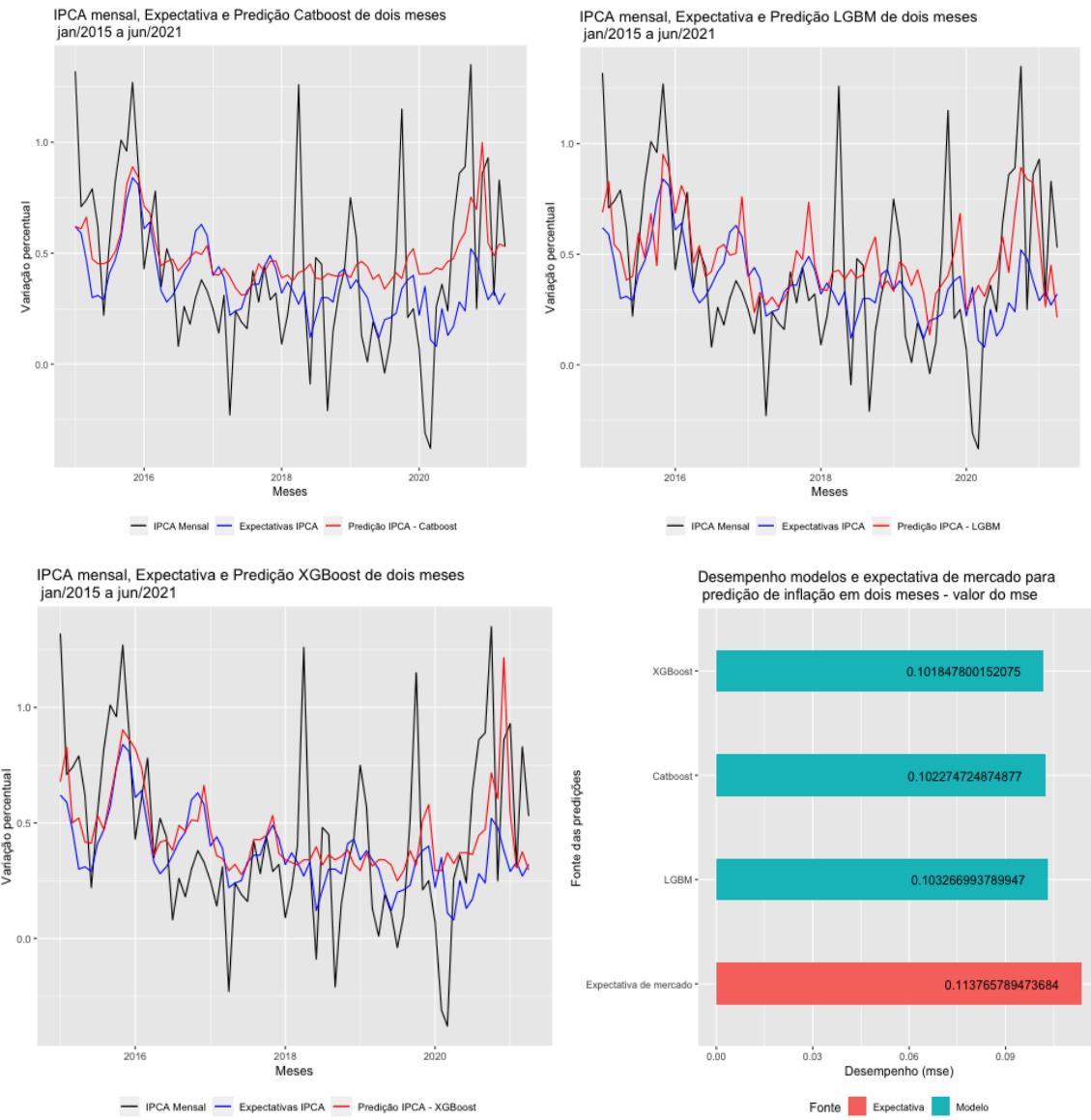


A predição da inflação de um mês espera-se que seja a que tenha melhor performance por ter um espaços na série temporal mais curto, com variáveis próximas do evento que se quer prever. Com os cálculos feitos o erro quadrático médio, observou-se que de fato o erro dos modelos é menor se comparado com de outros lags. Para esse modelo de um mês, apenas o modelo de CatBoost teve um erro menor que a média da expectativa de mercado. No entanto, os outros dois modelos também performaram bem, pois a diferença foi e apenas um ponto percentual em relação ao CatBoost.



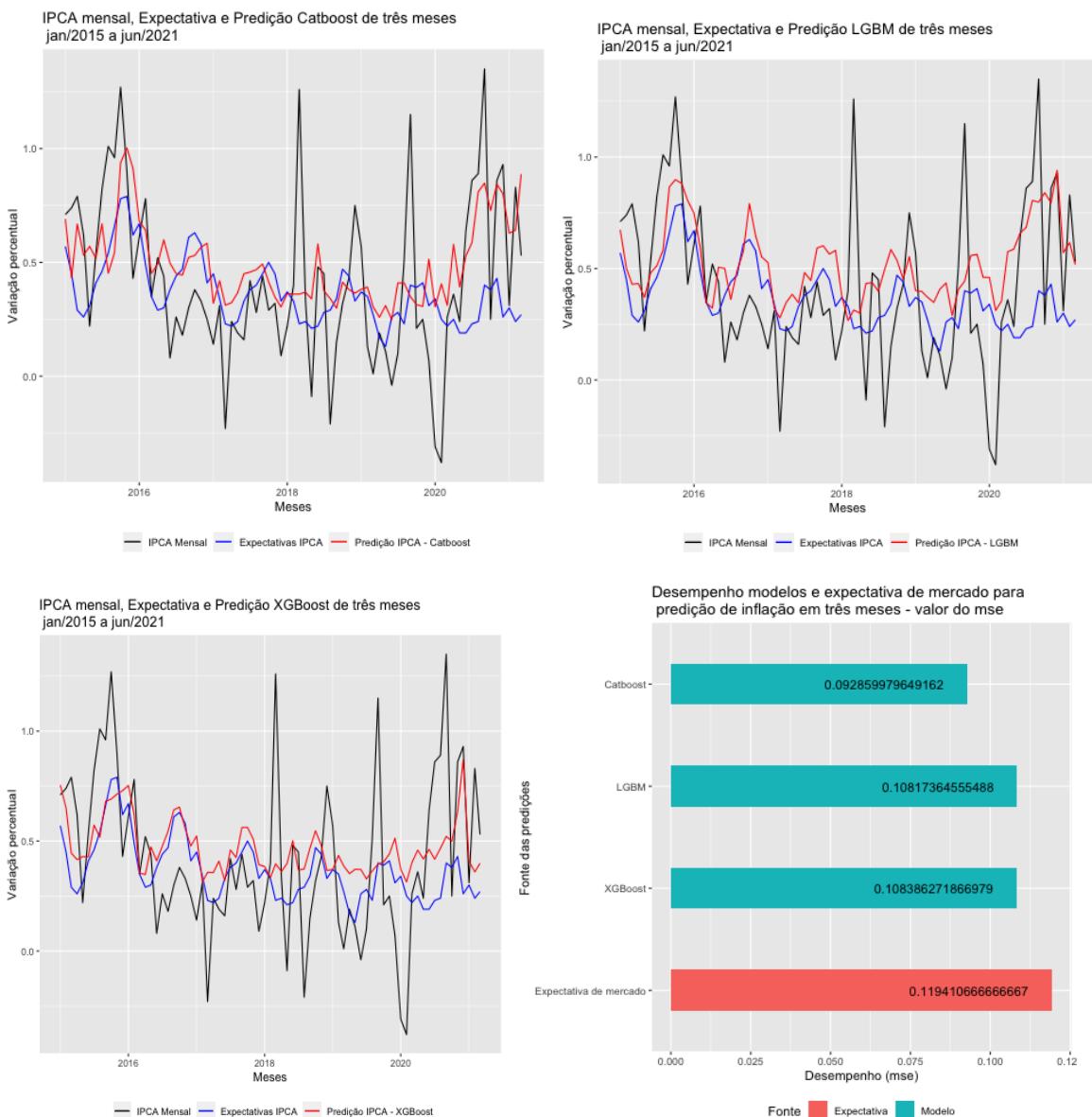
6.1.2 Predição de dois meses

Para previsão de dois meses, os três modelos tiveram comportamento e performance muito similares. O erro quadrático médio ficou praticamente iguais, com curvas ajustadas em situações muito parecidas.



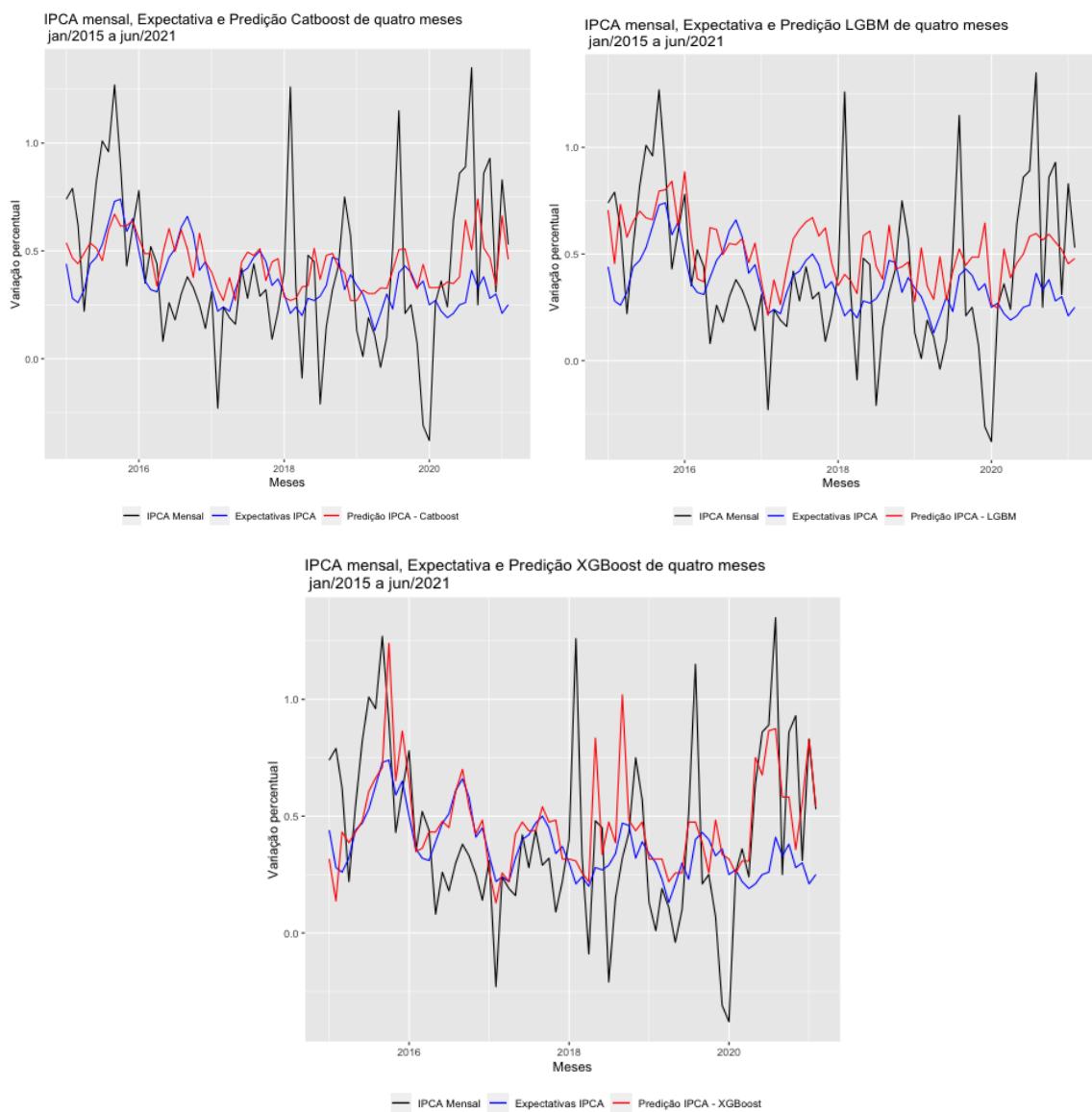
6.1.3 Predição de três meses

Assim como aconteceu com os modelos para previsão de um mês, os modelos referentes a três meses tiveram um bom ajuste à curva, com comportamentos semelhantes entre todos eles. Além disso, semelhante a outros *lags*, os modelos não identificam bem comportamento da inflação em situação de descenso do índice de inflação, o que ocorre em situações de quedas abruptas na atividade econômica.

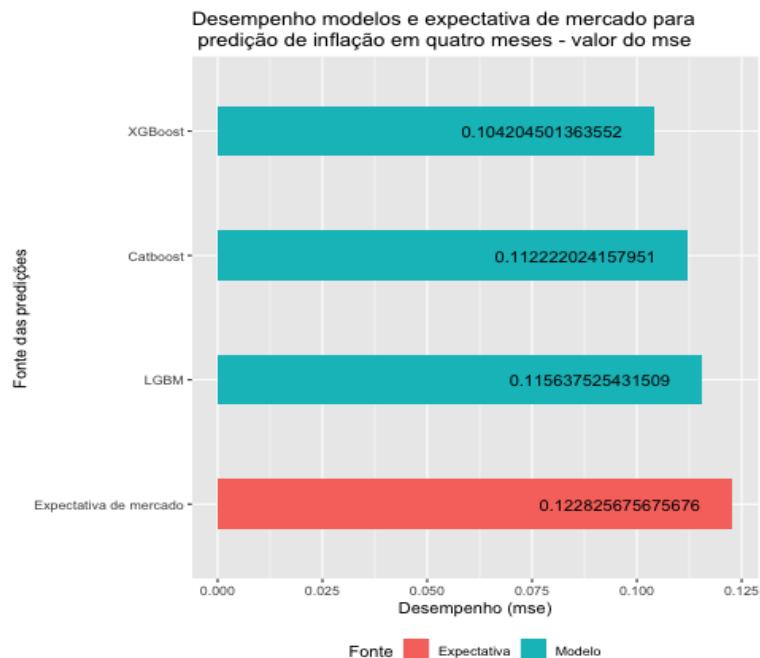


6.1.4 Predição de quatro meses

Por fim, para prever inflação com um horizonte temporal de 4 meses, espera-se que tenha uma performance pior e os modelos tenham mais de dificuldade de prever esses valores. No entanto, como podemos observar no modelo de XGBoost, a estratégia utilizada para prever foi apostar em uma maior variabilidade de valores de saída, para que pudesse captar justamente aqueles eventos que a inflação possa ser maior, o que tornou o modelo mais bem ajustado à curva.



Em se tratando dos modelos CatBoost e LightGBM, observou-se uma similaridade no comportamento e também na performance em termos de erros. XGBoost teve um erro quadrático médio melhor que os outros dois, sinalizando que pode ser um modelo que entregue melhores resultados para previsões com horizonte temporal mais alongado.



Em suma, identificou-se que performance dos modelos se sobressaiu a maior parte das vezes às expectativas de mercado, principalmente em horizontes de tempo mais longos. Ainda fica o desafio de melhor performance para eventos adversos, como pandemias e crises abruptas em geral, que tem impacto inflacionário e precisa de uma forma de identificação via dados para poder prever esses momentos.

6.2. Importância das variáveis

Com objetivo de dar mais transparências nas decisões dos modelos aqui listados, buscou-se formas de extrair informações dos outputs dos modelos para auxiliar na tomada de decisão. Para este trabalho especificamente se extraiu informações de importância de variáveis, até mesmo pra entender quais tem o peso maior na predição em comparação a outras.

Para isso, utilizou-se da biblioteca chamada SHAP, que explica saída dos modelos através da abordagem da teoria dos jogos. Esta teoria, concebida por Jhon Nash, prêmio nobel da economia e ex-professor do Departamento de Matemática de Princeton, modelou a forma de iteração entre atores que concebe suas ações com base no comportamento do outro, e se abriu um novo de estudos para diversas áreas.

No caso deste trabalho, a função da biblioteca é gerar *shap values*, que são valores da contribuição marginal média da variável se comparada às outras para o resultado em questão. Com esses valores, é possível identificar quais variáveis tem contribuído mais para os resultados gerados nos modelos.

O script que cria os gráficos da importância das variáveis é o da imagem abaixo. Além disso, cria-se tabelas no banco de dados local das predições.

```
conn = sqlite3.connect("../data/app_db.db")

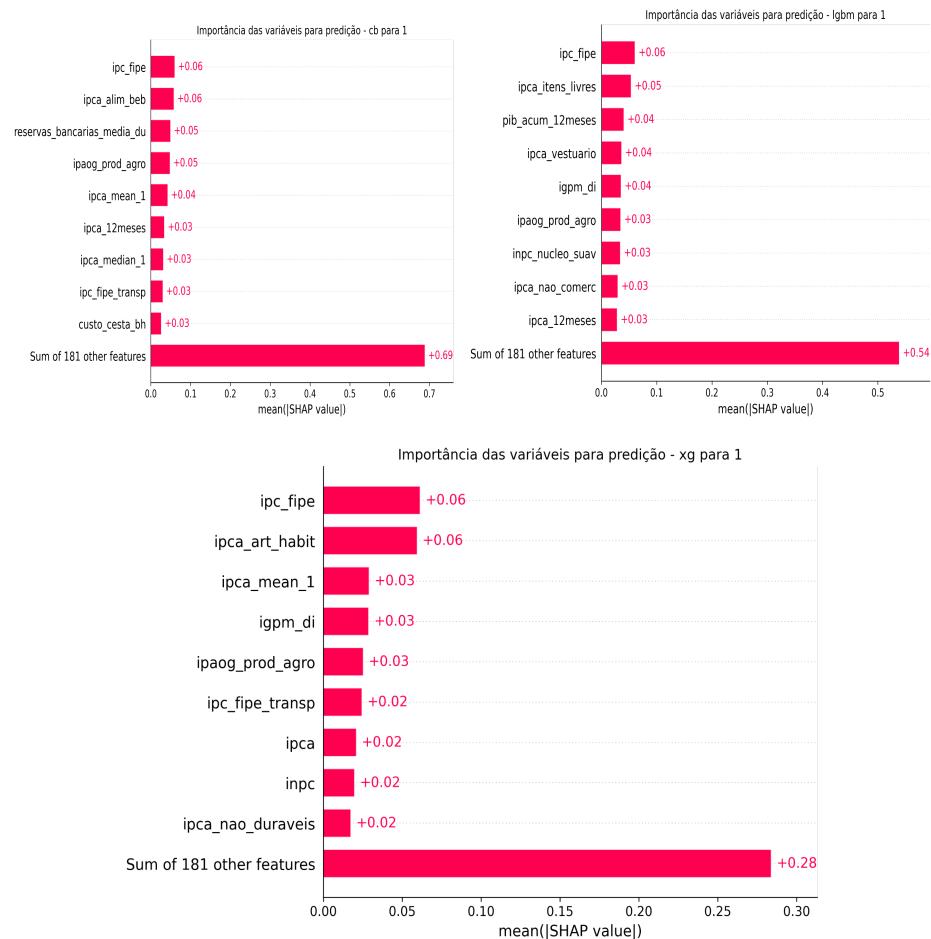
for l in [1, 2, 3, 4]:
    dados = criar_dataset(l)
    preds = dados[[f'y{l}', f'ipca_mean_{l}']].iloc[240:]
    train = dados.iloc[:240]
    test = dados.iloc[240:]
    xtr, xts = np.array(train.drop(["y1", "y2", "y3", "y4", "y5", "y6"], axis=1)), np.array(test.drop(["y1", "y2", "y3", "y4", "y5", "y6"], axis=1))
    ytr, yts = np.array(train[f'y{l}']), np.array(test[f'y{l}'])

    for i in ["cb", "xg", "lgbm"]:
        mdl = pickle.load(open(f"../models/saved.mdl_ipca_{i}_y{l}.pkl", 'rb'))
        mdl.fit(xtr, ytr)
        preds[f'y_preds_{i}_{l}'] = mdl.predict(xts)

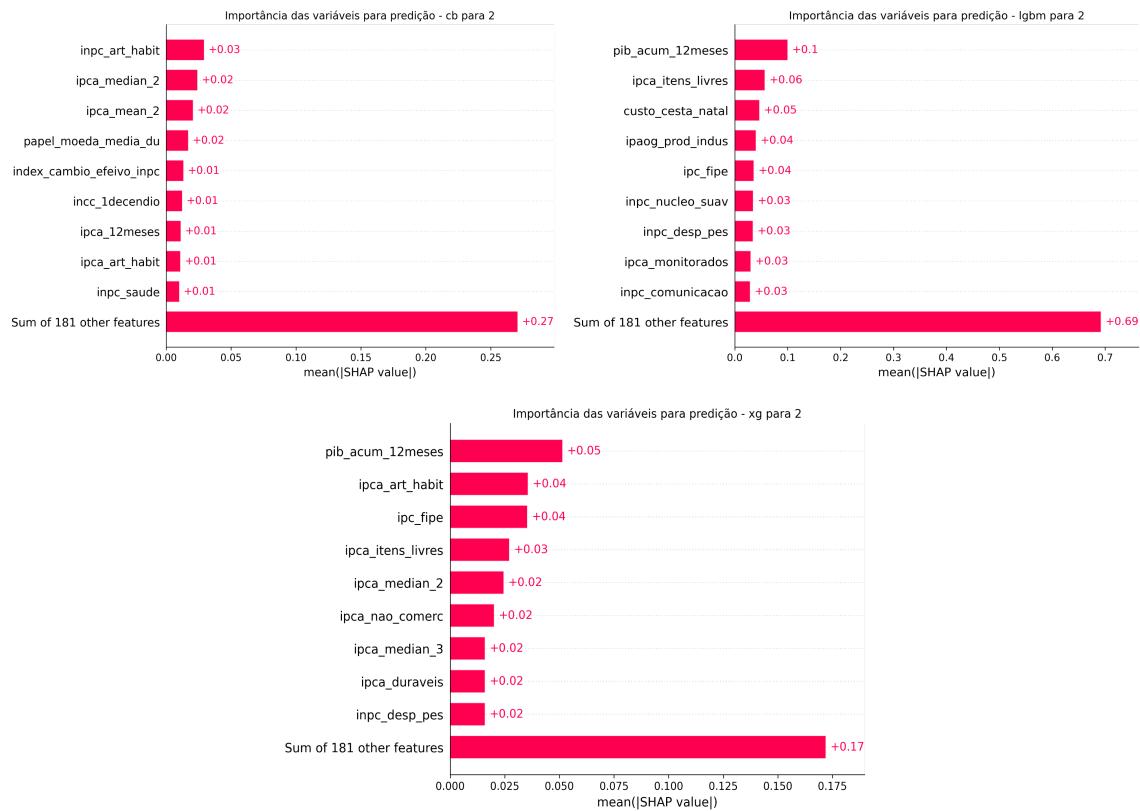
        mdl = pickle.load(open(f"../models/saved.mdl_ipca_{i}_y{l}.pkl", 'rb'))
        mdl.fit(train.drop(["y1", "y2", "y3", "y4", "y5", "y6"], axis=1), train[f'y{l}'])
        explainer = shap.TreeExplainer(mdl)
        shap_values = explainer(train.drop(["y1", "y2", "y3", "y4", "y5", "y6"], axis=1))
        shap.plots.bar(shap_values, show=False)
        plt.title(f'Importância das variáveis para predição - {i} para {l}')
        plt.savefig(f"../plots/plots_explain/mdl_explain_{i}_{l}.png", dpi=300, bbox_inches = "tight")
        plt.close()

    preds.to_sql(f"preds_l{l}", conn, if_exists="replace")
```

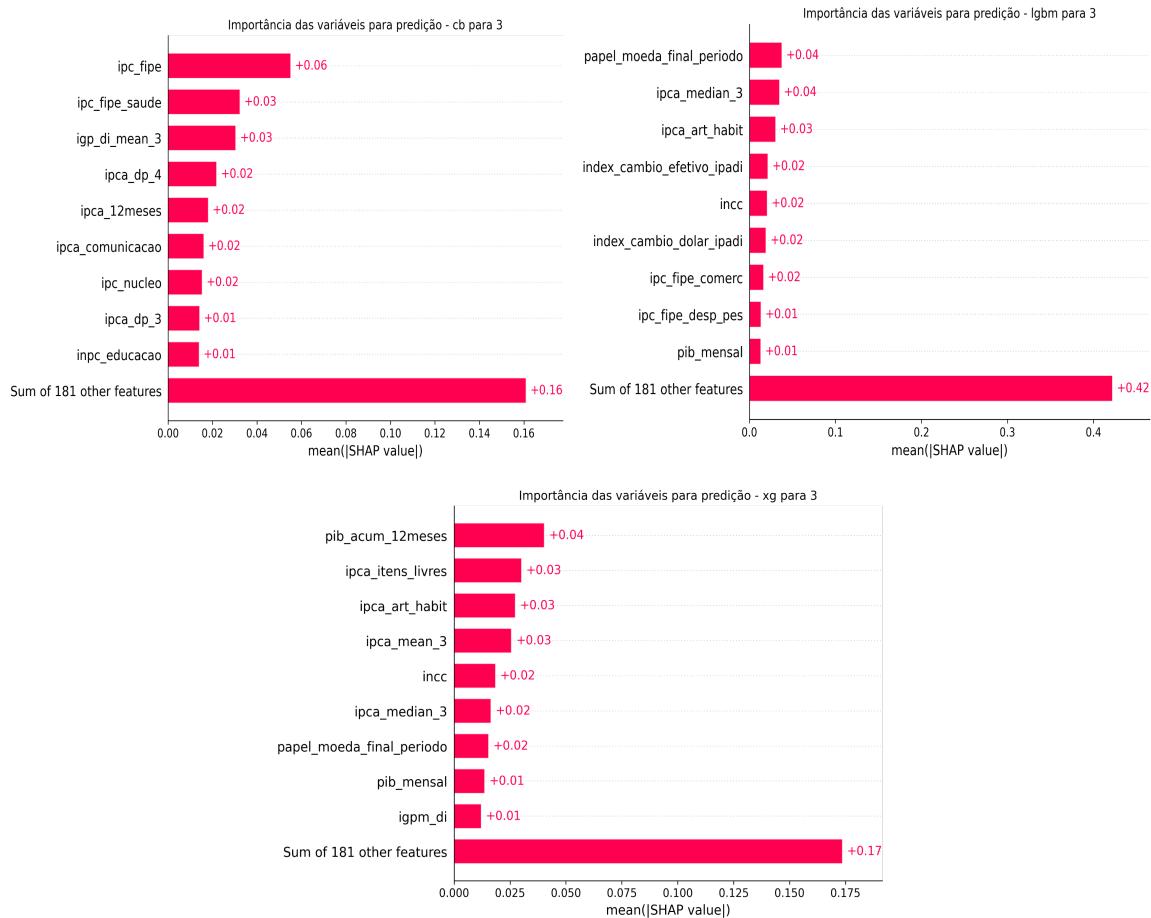
No que se refere a importância das variáveis para um mês de previsão, observa-se que os três algoritmos tiveram mesma a variável no topo do ranking: ipc_fipe. No entanto, a colocação das outras variáveis é distinta em todos os algoritmos, o que mostra que tomam decisões diversas, apesar da similaridade nos resultados.



Em se tratando das previsões para dois meses, temos um cenário de importância de variáveis diferente do anterior. O PIB (Produto Interno Bruto), por exemplo, acabou se destacando em dois algoritmos, LGBM e XGBoost, e entrou o índice de preços como INPC de utensílios para habitação como variável preditora para CatBoost, como o IPCA dos mesmos produtos também entrou no XGBoost.

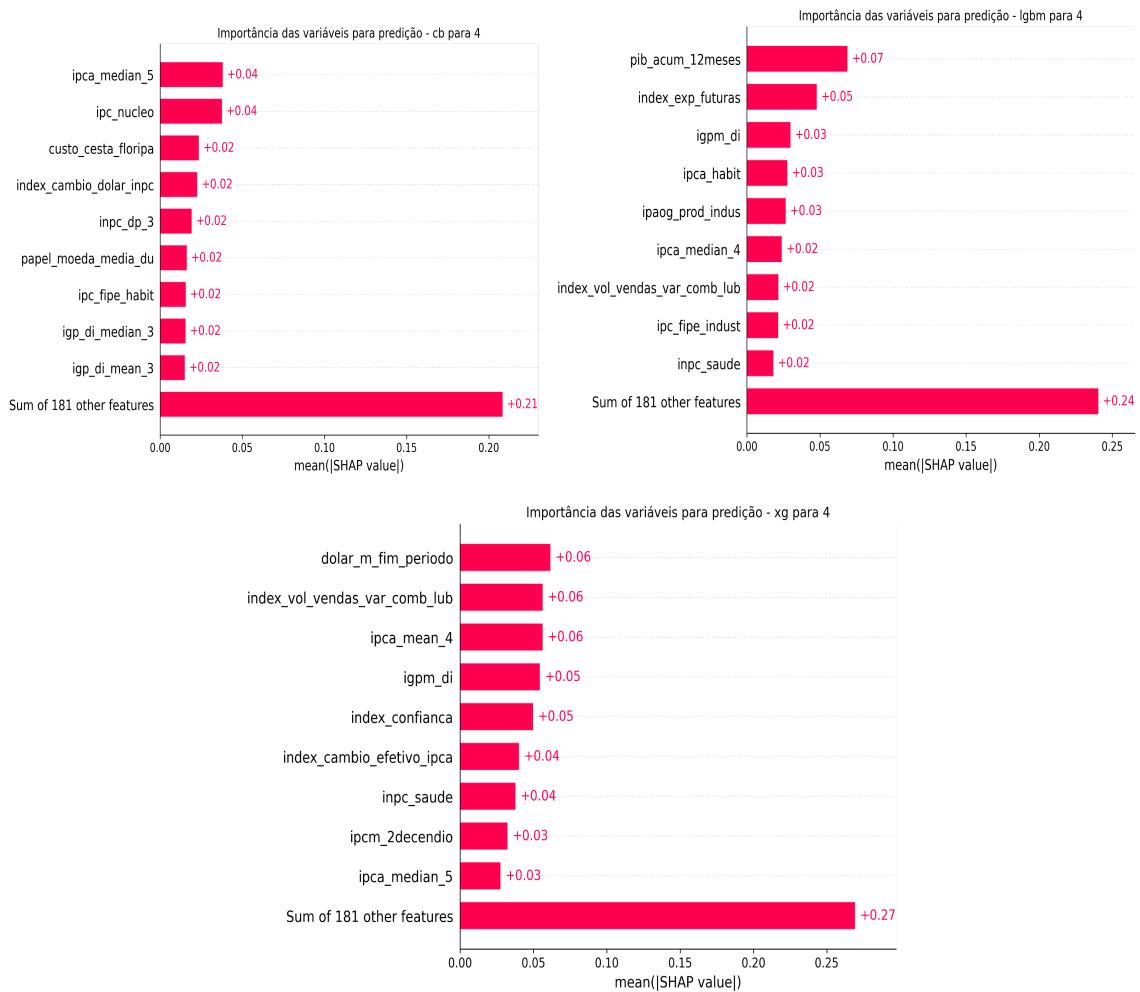


Para a previsão da inflação de três meses, novamente as variáveis são diferentes das identificadas anteriormente, com diferença do PIB acumulado de 12 meses para os modelos de XGBoost, que também é uma variável importante para previsão em um lag de dois. Além disso, o IPC da FIPE aparece novamente no CatBoost, a variável mais importante dos três algoritmos para previsão de inflação de um mês.



Por fim, para a previsão da inflação de quatro meses, o nível de atividade econômica apareceu mais uma vez como uma variável importante, porém para o algoritmo LGBM, além do index de expectativas futuras. Além disso, outra variável que surpreendeu foi o dólar mensal de fim de período, variável inclusa para captar variações no câmbio, que parece ser uma informação importante para o XGBoost prever inflação de quatro meses.

Outra informação importante nestas previsões de quatro meses é a inclusão da mediana de expectativa de inflação para cinco meses, que é a variável mais importante para o Catboost. A inclusão de variáveis de cinco meses, e não apenas quatro, deve-se ao fato de tentar captar não só a previsão do mercado, mas tendência de mudança, e acabou sendo uma variável importante para o modelo.



8. Links

Link para o vídeo: <https://youtu.be/60bBd9XuOyg>

Link para o repositório: https://github.com/pbrazil/predict_ipca

REFERÊNCIAS

TERRA, L. A. A.; PASSADOR, J. L.. **Redes neurais artificiais na previsão da inflação: aplicação como ferramenta de apoio à análise de decisões financeiras em organizações de pequeno porte.** São Paulo, mar/2012.

RIBEIRO, A.. **O samba dos juros.** Valor Econômico, 2013. Disponível em: <<https://valor.globo.com/eu-e/coluna/o-samba-dos-juros.ghtml>>. Acesso em: 12 de jul. de 2021.

BRESSER-PEREIRA, L. C.; NAKANO, Y. **Fatores aceleradores, mantenedores e sancionadores da inflação.** Revista de Economia Política, v. 4, n. 1, p. 5-21, jan./mar. 1984.

SHWARTZ-ZIV, R.; ARMON, A.. Tabular Data: Deep Learning is Not All You Need. jun, 2021.

KAUL, P.; GOLOVIN, D.; KOCHANSKI, G.. **Hyperparameter tuning in Cloud Machine Learning Engine using Bayesian Optimization.** Agosto, 2017. Acesso em: <<https://cloud.google.com/blog/products/ai-machine-learning/hyperparameter-tuning-cloud-machine-learning-engine-using-bayesian-optimization>>

MANDOT, P.. **O que é LightGBM, como implementá-lo? Como ajustar os parâmetros?.** Agosto, 2018. Acesso em: <https://ichi.pro/pt/o-que-e-lightgbm-como-implementa-lo-como-ajustar-os-parametros-105778469517308>

MASSARON, L.; MARINELLI, P.. **Kaggle Days Paris - Skopt + CatBoost solution.** Setembro, 2019. Acesso em: <https://www.kaggle.com/lucamassaron/kaggle-days-paris-skopt-catboost-solution>

GANDHI, S. A.. Tuning XGBoost Hyperparameters with Scikit Optimize. Agosto, 2019. Acesso em: <https://towardsdatascience.com/how-to-improve-the-performance-of-xgboost-models-1af3995df8ad>

APÊNDICE

| Nome completo | Nome da coluna/variável | Código SGS | Período | Início |
|--|-------------------------|------------|---------|------------|
| | | | cidade | |
| Índice de Expectativas Futuras | index_exp_futuras | 4395 | M | 31/03/1999 |
| Índice de Confiança do Consumidor | index_confianca | 4393 | M | 31/03/1999 |
| Índice de Condições Econômicas Atuais | index_cond_econ_atuais | 4394 | M | 31/03/1999 |
| Índice de Atividade Econômica do Banco Central - IBC-Br | ibc_br | 24363 | M | 01/01/2003 |
| NFSP sem desvalorização cambial - Fluxo mensal corrente - Resultado primário - Total - Setor público consolidado | nfsp_rp | 4649 | M | 01/12/2001 |
| Índice nacional de preços ao consumidor-Amplº (IPCA) - Comercializáveis | ipca_comerc | 4447 | M | 01/01/1992 |
| Índice de Preços ao Consumidor-Amplº (IPCA) - Não comercializáveis | ipca_nao_comerc | 4448 | M | 01/01/1992 |
| Índice nacional de preços ao consumidor - Amplº (IPCA) - Itens livres | ipca_itens_livres | 11428 | M | 31/01/1992 |
| Índice Nacional de Preços ao Consumidor - Amplº (IPCA) - Serviços | ipca_servicos | 10844 | M | 31/01/1992 |
| Índice de Preços ao Consumidor-Amplº (IPCA) - Duráveis | ipca_duraveis | 10843 | M | 31/01/1992 |
| Índice de Preços ao Consumidor-Amplº (IPCA) - Bens semi-duráveis | ipca_bens_semidur | 10842 | M | 31/01/1992 |
| Índice de Preços ao Consumidor-Amplº (IPCA) - Bens não-duráveis | ipca_nao_duraveis | 10841 | M | 31/01/1992 |
| Índice Nacional de Preços ao Consumidor Amplº (IPCA) - Índice de difusão | inpc_index_dif | 21379 | M | 31/08/1999 |
| Índice nacional de preços ao consumidor-Amplº (IPCA) - Núcleo médias apuradas com suavização | inpc_nucleo_suav | 4466 | M | 31/01/1996 |
| Índice geral de preços do mercado (IGP-M) | igpm | 189 | M | 30/06/1989 |
| Índice geral de preços-disponibilidade interna (IGP-DI) | igpm_di | 190 | M | 29/02/1944 |
| Índice de preços ao consumidor-Brasil (IPC-Br) | ipc_br | 191 | M | 31/01/1990 |
| Índice Nacional de Custo da Construção-Mercado (INCC-M) | incc | 7456 | M | 30/09/1994 |
| Índice de preços ao produtor amplo | ipa | 225 | M | 29/02/1944 |
| Índice de Preços ao Consumidor-Brasil (IPC-Br) - Núcleo | ipc_nucleo | 4467 | M | 31/01/ |

| | | | | | |
|---|------------------------|------|---|----------------|------|
| | | | | | 1996 |
| Índice Geral de Preços-10 (IGP-10) | igp10 | 7447 | M | 30/09/ 1994 | |
| Índice Geral de Preços-Mercado (IGP-M) - 1º Decêndio | igpm_1decendio | 7448 | M | 31/08/ 1994 | |
| Índice Geral de Preços-Mercado (IGP-M) - 2º Decêndio | igpm_2decendio | 7449 | M | 31/08/ 1994 | |
| Índice de Preços por Atacado-Mercado (IPA-M) | ipam | 7450 | M | 30/09/ 1994 | |
| Índice de Preços por Atacado-Mercado (IPA-M) - 1º Decêndio | ipam_1decendio | 7451 | M | 31/10/ 1994 | |
| Índice de Preços por Atacado-Mercado (IPA-M) - 2º Decêndio | ipam_2decendio | 7452 | M | 31/10/ 1994 | |
| Índice de Preços ao Consumidor-Mercado (IPC-M) | ipcm | 7453 | M | 30/09/ 1994 | |
| Índice de Preços ao Consumidor-Mercado (IPC-M) - 1º Decêndio | ipcm_1decendio | 7454 | M | 31/10/ 1994 | |
| Índice de Preços ao Consumidor-Mercado (IPC-M) - 2º Decêndio | ipcm_2decendio | 7455 | M | 31/10/ 1994 | |
| Índice Nacional de Custo da Construção-Mercado (INCC-M) - 1º Decêndio | incc_1decendio | 7457 | M | 31/10/ 1994 | |
| Índice Nacional de Custo da Construção-Mercado (INCC-M) - 2º Decêndio | incc_2decendio | 7458 | M | 31/10/ 1994 | |
| Índice de Preços por Atacado-Oferta Global (IPA-OG) - Produtos industriais | ipaog_prod_indus | 7459 | M | 31/01/ 1994 | |
| Índice de Preços por Atacado-Oferta Global (IPA-OG) - Produtos agrícolas | ipaog_prod_agro | 7460 | M | 31/01/ 1994 | |
| Índice nacional de preços ao consumidor (INPC) | inpc | 188 | M | 30/04/ 1979 | |
| Índice de preços ao consumidor-São Paulo (IPC-Fipe) | ipc_fipe | 193 | M | 28/02/ 1939 | |
| Índice de preços ao consumidor (IPC-Fipe) - Segunda quadrissemana | ipc_fipe_2quadrisemana | 272 | M | 31/01/ 1991 | |
| Índice nacional de preços ao consumidor-amplio (IPCA) | ipca | 433 | M | 02/01/ 1980 | |
| Índice nacional de preços ao consumidor-Amplio (IPCA) - Alimentação e bebidas | ipca_alim_beb | 1635 | M | 31/01/ 1991 | |
| Índice nacional de preços ao consumidor-Amplio (IPCA) - Habitação | ipca_habit | 1636 | M | 31/01/ 1991 | |
| Índice nacional de preços ao consumidor-Amplio (IPCA) - Artigos de residência | ipca_art_habit | 1637 | M | 31/01/ 1991 | |
| Índice nacional de preços ao consumidor-Amplio (IPCA)- Vestuário | ipca_vestuario | 1638 | M | 31/01/ 1991 | |
| Índice de nacional de preços ao consumidor-Amplio (IPCA) - Transportes | ipca_transportes | 1639 | M | 31/08/ 1999 | |

| | | | | |
|--|------------------------|------|---|----------------|
| Índice nacional de preços ao consumidor-Amplo (IPCA) - Comunicação | ipca_comunicacao | 1640 | M | 31/08/ 1999 |
| Índice nacional de preços ao consumidor-Amplo (IPCA) - Saúde e cuidados pessoais | ipca_saude | 1641 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor-Amplo (IPCA) - Despesas pessoais | ipca_desp_pes | 1642 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor-Amplo (IPCA) - Educação | ipca_educacao | 1643 | M | 31/08/ 1999 |
| Índice nacional de preços ao consumidor (INPC) - Alimentação e bebidas | inpc_alim_beb | 1644 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Habitação | inpc_habit | 1645 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Artigos de residência | inpc_art_habit | 1646 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Vestuário | inpc Vestuario | 1647 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Transporte | inpc_transporte | 1648 | M | 31/08/ 1999 |
| Índice nacional de preços ao consumidor (INPC) - Comunicação | inpc_comunicacao | 1649 | M | 31/08/ 1999 |
| Índice nacional de preços ao consumidor (INPC) - Saúde e cuidados pessoais | inpc_saude | 1650 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Despesas pessoais | inpc_desp_pes | 1651 | M | 31/01/ 1991 |
| Índice nacional de preços ao consumidor (INPC) - Educação | inpc_educacao | 1652 | M | 31/08/ 1999 |
| Índice Nacional de Preços ao Consumidor - Amplo (IPCA) - Monitorados | ipca_monitorados | 4449 | M | 01/01/ 1992 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Primeira quadrissemana | ipc_fipe_1quadrisemana | 7463 | M | 31/01/ 1995 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Terceira quadrissemana | ipc_fipe_3quadrisemana | 7464 | M | 31/01/ 1995 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Alimentação | ipc_fipe_aliment | 7465 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Industrializados | ipc_fipe_indust | 7467 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - In natura | ipc_fipe_innatura | 7468 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Habitação | ipc_fipe_habit | 7469 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Transportes | ipc_fipe_transp | 7470 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Despesas pessoais | ipc_fipe_desp_pes | 7471 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Vestuário | ipc_fipe_vest | 7472 | M | 31/01/ 1997 |

| | | | | |
|---|-----------------------|-------|---|----------------|
| Índice de Preços ao Consumidor (IPC-Fipe) - Saúde | ipc_fipe_saude | 7473 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Educação | ipc_fipe_educacao | 7474 | M | 31/01/ 1997 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Comercializáveis | ipc_fipe_comerc | 7475 | M | 31/01/ 1995 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Não-comercializáveis | ipc_fipe_nao_comerc | 7476 | M | 31/01/ 1995 |
| Índice de Preços ao Consumidor (IPC-Fipe) - Monitorados | ipc_fipe_monit | 7477 | M | 31/01/ 1995 |
| Índice de Preços ao Consumidor-Amplo (IPCA) - 15 | ipca15 | 7478 | M | 31/05/ 2000 |
| Índice de Preços ao Consumidor-Amplo - Especial (IPCA-E) | ipca_e | 10764 | M | 31/01/ 1992 |
| Índice nacional de preços ao consumidor - amplo (IPCA) - em 12 meses | ipca_12meses | 13522 | M | 01/12/ 1980 |
| Índice Nacional de Preços ao Consumidor - Amplo (IPCA) - Industriais | ipca_industriais | 27863 | M | 01/08/ 1999 |
| Indice Nacional de Preços ao Consumidor - Amplo (IPCA) - Alimentação no domicílio | ipca_alim_dom | 27864 | M | 01/08/ 1999 |
| Cesta básica - Aracaju | custo_cesta_aracaju | 7479 | M | 31/01/ 1998 |
| Cesta básica - Belém | custo_cesta_belem | 7480 | M | 31/01/ 1998 |
| Cesta básica - Belo Horizonte | custo_cesta_bh | 7481 | M | 31/01/ 1998 |
| Cesta básica - Brasília | custo_cesta_brasilia | 7482 | M | 31/01/ 1998 |
| Cesta básica - Curitiba | custo_cesta_curitiba | 7483 | M | 31/01/ 1998 |
| Cesta básica - Florianópolis | custo_cesta_floripa | 7484 | M | 31/01/ 1998 |
| Cesta básica - Fortaleza | custo_cesta_fortaleza | 7485 | M | 31/01/ 1998 |
| Cesta básica - Goiânia | custo_cesta_goiania | 7486 | M | 31/01/ 1998 |
| Cesta básica - João Pessoa | custo_cesta_jp | 7487 | M | 31/01/ 1998 |
| Cesta básica - Natal | custo_cesta_natal | 7488 | M | 31/01/ 1998 |
| Cesta básica - Porto Alegre | custo_cesta_poa | 7489 | M | 31/01/ 1998 |
| Cesta básica - Recife | custo_cesta_recife | 7490 | M | 31/01/ 1998 |
| Cesta básica - Rio de Janeiro | custo_cesta_rj | 7491 | M | 31/01/ 1998 |

| | | | | |
|--|----------------------|------|---|----------------|
| Cesta básica - Salvador | custo_cesta_salvador | 7492 | M | 31/01/ 1998 |
| Cesta básica - São Paulo | custo_cesta_sp | 7493 | M | 31/01/ 1998 |
| Cesta básica - Vitória | custo_cesta_vitoria | 7494 | M | 31/01/ 1998 |
| PIB acumulado dos últimos 12 meses - Em US\$ milhões | pib_acum_12meses | 4192 | M | 31/01/ 1990 |
| PIB mensal - Valores correntes (R\$ milhões) | pib_mensal | 4380 | M | 31/01/ 1990 |
| PIB acumulado dos últimos 12 meses - Valores correntes (R\$ milhões) | pib_acum_ult12meses | 4382 | M | 31/01/ 1990 |