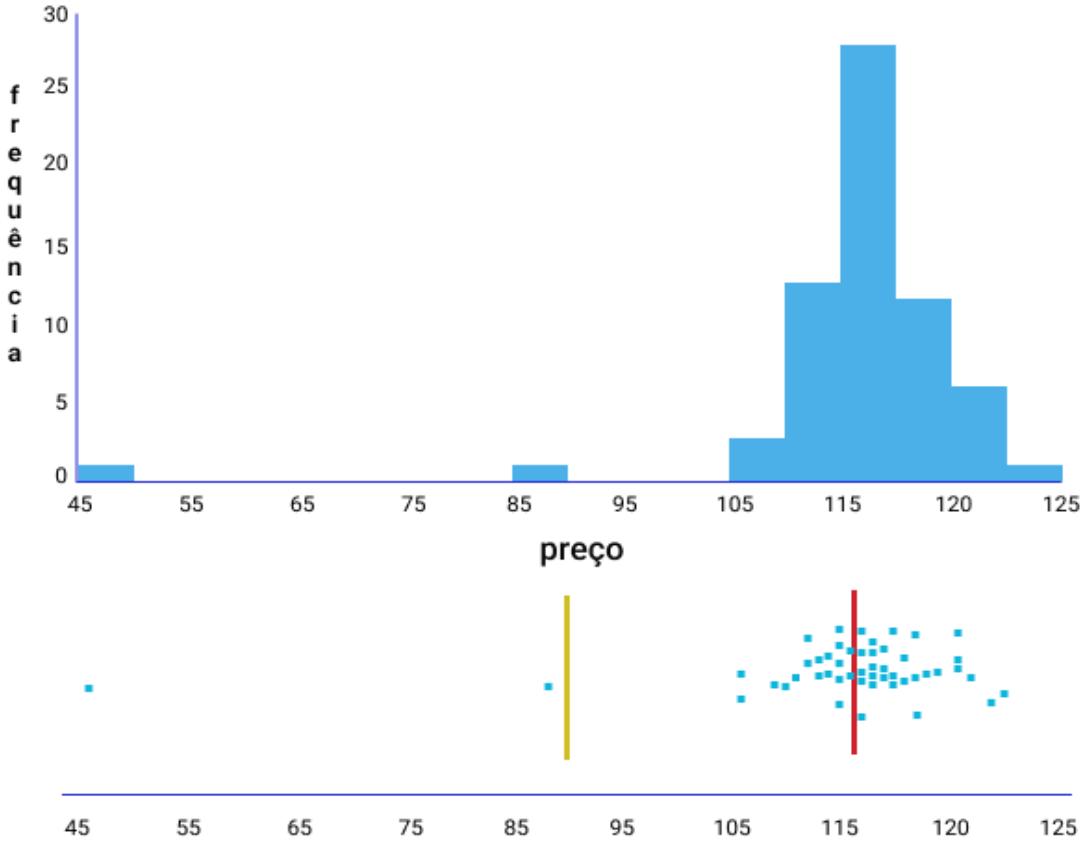
Detecção de anomalias de preços

Desafio americanas

Baseline


30
25
20
15
10
5
0
frequência
45
55
65
75
85
95
105
115
120
125
preço
45
55
65
75
85
95
105
115
120
125

MVP

```
def get_price_alert(id_oferta, date, limiar_inf, limiar_sup, new_price):
  mean = valid_data.loc[(id_oferta, date), ('preco_da_oferta', 'mean')][0]
  std = valid_data.loc[(id_oferta, date), ('preco_da_oferta', 'std')][0]
  if new_price > mean + limiar_sup * std:
    return True
  elif new_price < mean - limiar_inf * std:
    return True
  else:
    return False
```

MVP

```
# Exemplo
id_oferta = 'FfNzoXjebpoheVxG8D+AbqeQH9qfLtdS0l2zjcU7kvU='
date = '2021-07-20 00:00:00+00:00'
limiar_inf = 2.8
limiar_sup = 2.8
new_price = 180
get_price_alert(id_oferta, date, limiar_inf, limiar_sup, new_price)
```

```
True
```

```
# Exemplo
id_oferta = 'FfNzoXjebpoheVxG8D+AbqeQH9qfLtdS0l2zjcU7kvU='
date = '2021-07-20 00:00:00+00:00'
limiar_inf = 2.8
limiar_sup = 2.8
new_price = 125
get_price_alert(id_oferta, date, limiar_inf, limiar_sup, new_price)
```

```
False
```

Pontos de melhoria

- Teste de falso positivos
- Entrada de data por intervalo
- Trocar std por zscore
- Usar Ensemble methods(EM)
- Detecção de anomalia do Pycaret para(EM)
- Isolation Forest para o(EM)