**股票选股强度验证**

# 2021-12-16 16:52 选股测试

## 参数配置

测试方法描述：选股测试

测试起止日期：20190101 ~ 20211114

信号计算函数：  
def get\_signals(c: analyze.CZSC) -> OrderedDict:  
 """在 CZSC 对象上计算选股信号  
  
 :param c: CZSC 对象  
 :return: 信号字典  
 """  
 freq: Freq = c.freq  
 s = OrderedDict({"symbol": c.symbol, "dt": c.bars\_raw[-1].dt, "close": c.bars\_raw[-1].close})  
  
 default\_signals = [  
 # 以下是技术指标相关信号  
 Signal(k1=str(freq.value), k2="MA5状态", v1="其他", v2='其他', v3='其他'),  
 Signal(k1=str(freq.value), k2="KDJ状态", v1="其他", v2='其他', v3='其他'),  
 Signal(k1=str(freq.value), k2="MACD状态", v1="其他", v2='其他', v3='其他'),  
 Signal(k1=str(freq.value), k2="倒0笔", k3="潜在三买", v1="其他", v2='其他', v3='其他'),  
 ]  
 for signal in default\_signals:  
 s[signal.key] = signal.value  
  
 if not c.bi\_list:  
 return s  
  
 if len(c.bars\_raw) > 30 and c.freq in [Freq.W, Freq.M]:  
 if kdj\_gold\_cross(c.bars\_raw, just=False):  
 v = Signal(k1=str(freq.value), k2="KDJ状态", v1="金叉")  
 s[v.key] = v.value  
  
 if len(c.bars\_raw) > 100 and c.freq == Freq.D:  
 close = np.array([x.close for x in c.bars\_raw[-100:]])  
 ma5 = SMA(close, timeperiod=5)  
 if c.bars\_raw[-1].close >= ma5[-1]:  
 v = Signal(k1=str(freq.value), k2="MA5状态", v1="收盘价在MA5上方", v2='')  
 s[v.key] = v.value  
 if ma5[-1] > ma5[-2] > ma5[-3]:  
 v = Signal(k1=str(freq.value), k2="MA5状态", v1='收盘价在MA5上方', v2="向上趋势")  
 s[v.key] = v.value  
  
 diff, dea, macd = MACD(close, fastperiod=12, slowperiod=26, signalperiod=9)  
 if diff[-3:-1].mean() > 0 and dea[-3:-1].mean() > 0 and macd[-3] < macd[-2] < macd[-1]:  
 v = Signal(k1=str(freq.value), k2="MACD状态", v1="DIFF大于0", v2='DEA大于0', v3='柱子增大')  
 s[v.key] = v.value  
  
 # 倒0笔潜在三买  
 if c.freq == Freq.D and len(c.bi\_list) >= 5:  
 if c.bi\_list[-1].direction == Direction.Down:  
 gg = max(c.bi\_list[-1].high, c.bi\_list[-3].high)  
 zg = min(c.bi\_list[-1].high, c.bi\_list[-3].high)  
 zd = max(c.bi\_list[-1].low, c.bi\_list[-3].low)  
 else:  
 gg = max(c.bi\_list[-2].high, c.bi\_list[-4].high)  
 zg = min(c.bi\_list[-2].high, c.bi\_list[-4].high)  
 zd = max(c.bi\_list[-2].low, c.bi\_list[-4].low)  
  
 if zg > zd:  
 k1 = str(freq.value)  
 k2 = "倒0笔"  
 k3 = "潜在三买"  
 v = Signal(k1=k1, k2=k2, k3=k3, v1="构成中枢")  
 if gg \* 1.1 > min([x.low for x in c.bars\_raw[-3:]]) > zg > zd:  
 v = Signal(k1=k1, k2=k2, k3=k3, v1="构成中枢", v2="近3K在中枢上沿附近")  
 if max([x.high for x in c.bars\_raw[-7:-3]]) > gg:  
 v = Signal(k1=k1, k2=k2, k3=k3, v1="构成中枢", v2="近3K在中枢上沿附近", v3='近7K突破中枢GG')  
  
 if v and "其他" not in v.value:  
 s[v.key] = v.value  
  
 return s

事件具体描述：  
def get\_event():  
 event = Event(name="选股测试", operate=Operate.LO, factors=[  
 Factor(name="月线KDJ金叉\_日线MACD强势", signals\_all=[  
 Signal("月线\_KDJ状态\_任意\_金叉\_任意\_任意\_0"),  
 Signal('日线\_MACD状态\_任意\_DIFF大于0\_DEA大于0\_柱子增大\_0'),  
 Signal('日线\_MA5状态\_任意\_收盘价在MA5上方\_任意\_任意\_0'),  
 ]),  
  
 Factor(name="月线KDJ金叉\_日线潜在三买", signals\_all=[  
 Signal("月线\_KDJ状态\_任意\_金叉\_任意\_任意\_0"),  
 Signal('日线\_倒0笔\_潜在三买\_构成中枢\_近3K在中枢上沿附近\_近7K突破中枢GG\_0'),  
 Signal('日线\_MA5状态\_任意\_收盘价在MA5上方\_任意\_任意\_0'),  
 ]),  
 ])  
 return event