



# 量化金融分析师（AQF®）全国统一考试

## 模拟题

适用场次：2022 年 3 月

使用本模拟题，您应该遵守：

1. 本模拟题仅提供给参加 2022 年 3 月份 AQF 全国统一考试的考生，考生仅可以出于准备个人考试的目的查阅和打印本模拟题；
2. 严禁出于任何目的的复制、网络发布和传播、抄袭本模考题内容，如有违反，可能导致违纪或违法行为；

© 版权所有，侵权必究。

量化金融标准委员会

Standard Committee of Quantitative Finance

## 量化金融分析师（AQF®）全国统一考试模拟题

说明：本场考试中的代码都应采用 Python 3.X 版本作答。

解答题（每题 20 分，本部分共 100 分）

1. 李明，AQF，某量化基金经理，在量化研究的过程中，获取了 10 支股票的行业和量价信息存储在 SQLite 数据库 stock\_data 中。其中，表 stock\_price 存放这 10 支股票 2020 年的行情数据，部分数据如下：

Id	Date	Open	Close	High	Low	Volume	Code
0	2020-01-02	16.65	16.87	16.95	16.55	1530231.0	000001
1	2020-01-03	16.94	17.18	17.31	16.92	1116194.0	000001
2	2020-01-06	17.01	17.07	17.34	16.91	862083.0	000001

表 industry 存放相关行业的基本信息，部分数据如下：

Id	Industry	Description
1	financial service	Financial services include banking, securities, insurance, trust and fund
2	real estate	The real estate industry is an industry of real estate investment, development and management
3	internet	Internet industry is an industry based on computer network technology and using network platform to provide services

李明现希望在 SQLite 数据库 stock\_data 中创建第三张表 stock\_info，该表中存储 10 支股票的基本信息，包括股票代码（Code）、公司名称（CompanyName）、上市日期（ListingDate）、行业编号（IndustryId）四个字段，其中 Code、CompanyName、ListingDate 数据类型为文本类型，IndustryId 数据类型为整数型，并将 Code 设置为主键且非空。目前该信息存放在 csv 文件 stock\_info 中， csv 文件部分数据如下：

Code	CompanyName	ListingDate	IndustryId
000001	平安银行	2001-04-03	1

000002	万科 A	1991-01-29	2
601939	建设银行	2007-09-25	1

1.1. 请你在数据库 `stock_data` 中按照以上要求创建数据表 `stock_info`, 并将 csv 文件 `stock_info` 中的数据录入该表。请在 Python 环境下完成该任务。(8 分)

参考答案:

# 创建数据库连接 (2 分)

```
import sqlite3
```

```
import pandas as pd
```

```
conn = sqlite3.connect('stock_data.sqlite')
```

```
cur = conn.cursor()
```

# 创建数据表 (2 分)

```
sql = """
```

```
CREATE TABLE stock_info (
```

```
    Code TEXT PRIMARY KEY NOT NULL,
```

```
    CompanyName TEXT,
```

```
    ListingDate TEXT,
```

```
    IndustryId INTEGER
```

```
);
```

```
"""
```

```
cur.execute(sql)
```

```
cur.commit()
```

# 读取文件 (2 分)

```
stock_info = pd.read_csv('stock_info.csv', dtype={'Code':str},
encoding='utf-8')
```

# 录入数据 (2 分)

```
pd.io.sql.to_sql(stock_info, 'stock_info', con=conn, if_exists='replace',
index=False)
```

```
conn.commit()
```

1.2. 李明发现表 stock\_info 中数据有误，完美世界的上市日期应为 2011 年 10 月 28 日，误写为 2012 年 10 月 28 日，请你修改此项错误，只需写出 SQL 命令即可。（4 分）

参考答案：

```
UPDATE stock_info
SET ListingDate = "2011-10-28"
WHERE CompanyName = "完美世界";
```

1.3. 请查找出建设银行 2020 年 1 月 1 日到 2020 年 1 月 15 日的股票行情信息，只需写出 SQL 命令即可（4 分）

参考答案：

```
SELECT Date, Open, Close, High, Low
FROM stock_info AS s
JOIN stock_price AS p
ON s.Code = p.Code
WHERE CompanyName = "建设银行"
AND Date BETWEEN "2020-01-01" AND "2020-01-31"
```

1.4. 请打印出这 10 支股票收盘价的最大值（4 分）

参考答案：

```
SELECT Code, MAX(Close)
FROM stock_price
GROUP BY Code
```

2. 某研究员正在构建一个配对交易策略，他选取了两种期货指数作为配对交易的对象。为了验证此两种期货具有一定相关性，他获取此两种期货合约指数 2020 年的价格数据作为训练集，分别存储于 csv 文件 training1 和 training2 中，部分数据如下：

trade_date	close	trade_date	close
20191231	1080.609	20191231	1899.86
20191230	1073.342	20191230	1886.109

20191227	1072.131
20191226	1068.498

20191227	1885.051
20191226	1881.348

其中, `trade_date` 为交易日日期, `close` 为期货指数当日收盘价。同时, 他获取了 2021 年作为回测区间, 将 2021 年的期货指数价格数据存储于 csv 文件 `test` 中, 部分数据如下:

trade_date	close_1	close_2
2020/1/2	1086.633	1936.065
2020/1/3	1086.633	1931.162
2020/1/6	1083.801	1934.839
2020/1/7	1091.166	1929.936

2.1. 为了检验这两个期货指数是否适合进行配对交易, 该研究员准备使用训练集数据进行相关性检查, 请你使用 `training1` 和 `training2` 中的数据, 绘制两个期货指数的价格走势, 并计算秩相关系数。(8 分)

**参考答案:**

# 读取数据并进行拼接 (4 分)

```
training1 = pd.read_csv('training1.csv', index_col=0,
parse_dates=True).sort_index()
```

```
training2 = pd.read_csv('training2.csv', index_col=0,
parse_dates=True).sort_index()
training = training1.join(training2,
how='inner', lsuffix='_1', rsuffix='_2')
```

```
或 training = pd.merge(training1, training2, left_index=True,
right_index=True, suffixes=('_1', '_2'))
```

# 绘制价格走势 (2 分)

```
training.plot(figsize=(8, 6))
```

# 计算相关系数 (2 分)

```
training.corr(method='spearman')
```

2.2. 在检验了两个期货指数的相关性之后, 他认为这两种期货指数适合进行配对交易, 并准备使用 2021 年价格数据进行配对交易策略回测。该研究员认为, 当两个期货指数价差偏离均值 2 倍标准差时存在错误定价, 并进行开仓操作, 回归正常价格区间时进行平仓。请

你使用题目所提供的数据，计算交易信号。（9 分）

**参考答案：**

# 计算训练集均值和标准差（2 分）

```
training['spread'] = training['close_1'] - training['close_2']
```

```
mu = training['spread'].mean()
```

```
sigma = training['spread'].std()
```

# 读取测试集数据

```
test = pd.read_csv('test.csv', index_col=0, parse_dates=True)
```

# 计算测试集价差上下轨（4 分）

```
test['spread'] = test['close_1'] - test['close_2']
```

```
test['up'] = mu + 2 * sigma
```

```
test['down'] = mu - 2 * sigma
```

# 计算交易信号（3 分）

```
test['position_1'] = np.where(test['spread'] > test['up'], -1, 0)
```

```
test['position_1'] = np.where(test['spread'] < test['down'], 1,  
test['position_1'])
```

```
test['position_2'] = -test['position_1']
```

2.3. 请问构建配对交易策略的前提条件是什么？答出任意两点即可。（3 分）

**参考答案：**

“卖空机制”

作为市场中性策略（Market Neutral Strategy）的一种，没有卖空机制是不可能执行配对交易的。虽然中国 A 股市场已经有了融资融券机制，但标的证券的种类太少，没有转融通机制导致能借到证券的数量也不多，所以诸多限制使得配对交易并没有大量兴起。一些券商研究部的金融工程组撰写的报告大多是从理论上进行分析。相信这一现象会随着市场交易机制的完善而改变。

“构建配对组”

广义的配对交易范畴很大，只要有两个风险收益特征非常相似的证券都可以配对。比如，两份交割日不同的沪深 300 股指期货合约可以配对，四川长虹和它的权证可以配对，招商银行的 A 股和 H 股可以配对，同一行业内基本面相似的公司（比如建行和工行）可以配对。在

我们构建配对组时，我们既可以按照基本面来分析，比如寻找同一行业内主营业务同质化较高的公司。我们也可以完全按照统计原理进行筛选，寻找相关系数较高的股票。

#### “价差（Spread）收敛”

根据对历史数据的统计，两只股票的价差是长期稳定的，但稳定性会因投资者“追涨杀跌”的不理性交易而打破，导致价差扩大。这时我们就同时建立多空头寸，卖空相对高估的股票，买入相对低估的股票，等待价差收敛。价差收敛后，我们同时将多空头寸平仓，就可以获得收益。“价差收敛”是配对交易能否获利的关键假设，这种收敛的假设是根据过去配对股票价差的标准差来判断的，是数量分析的结果。如果某只公司的基本面发生了变化，价差就可能进一步扩大，此时只能清仓止损了。举一个极端的例子就是在 2008 年下半年将高盛和雷曼兄弟的股票配对。

3. 在多因子策略中，对于有效因子的挖掘与测试是模型构建的重要前提。除了使用分层检验之外，回归法和因子 IC 值也是衡量因子对收益的预测能力的方法之一。李明，某量化研究员，将股票池中的股票在 2021 年上半年的月度因子值数据和月度收益率数据，分别存储于 csv 文件 `factor.csv` 和 `returns.csv` 中。其中，`factor.csv` 包含股票代码（code）、交易日日期（trade\_date）、净资产收益率（roe）数据，部分数据如下：


code	trade_date	roe
300120.SZ	20200104	2.6246
603595.SH	20200104	6.2389
300218.SZ	20200104	7.0669
300338.SZ	20200104	10.8223

002222.SZ	,20210104	5.1984
-----------	-----------	--------

returns.csv 中包含股票代码(code)、交易日日期(trade\_date)、月度收益率(pct\_chg)数据, 部分数据如下:

code	trade_date	pct_chg
000001.SZ	20210123	-0.0553
000002.SZ	20210123	-0.0994
000004.SZ	20210123	0.0102
000005.SZ	20210123	-0.0388
000006.SZ	20210123	-0.0522

3.1. 李明想要使用题目中提供的数据, 筛选出 2021 年 4 月因子值数据和收益率数据, 使用最小二乘法回归分析, 判断在 95%显著性水平下, 该因子是否显著。另外, 在数据处理的过程中, 李明发现净资产收益率数据单位为%, 即数据放大了 100 倍, 将其调整为收益率同量纲对数据分析效果可能更好。请按照如上要求处理数据, 并完成上述分析 (10 分)

参考答案:

```
# 获取因子值数据并做基础数据处理 (4 分)
factor = pd.read_csv('factor.csv')
factor['trade_date'] = pd.to_datetime(factor['trade_date'],
format='%Y%m%d')
factor.set_index('trade_date', inplace=True)
factor['roe'] = factor['roe'] / 100
# 获取收益率数据并做基础数据处理 (1 分)
returns = pd.read_csv('returns.csv')
returns['trade_date'] = pd.to_datetime(returns['trade_date'],
format='%Y%m%d')
returns.set_index('trade_date', inplace=True)
# 回归分析 (5 分)
data = pd.merge(factor.loc['2021-04'], returns.loc['2021-04'])
import statsmodels.api as sm
```



```
x = data['roe']
x = sm.add_constant(x)
y = data['pct_chg']
est = sm.OLS(y, x).fit()
est.summary()
```

3.2 请你根据题目提供的数据，计算 2021 年该因子显著的比例，以及当因子显著时，因子对收益预测的方向（1 表示因子值越大收益越高，-1 表示因子值越大收益越低），效果如下（部分数据）：

	significant	direction
2021-04	True	1.0
2021-05	False	-1.0
2021-06	True	1.0

（10 分）参考答案：

```
result = pd.DataFrame(columns=['significant', 'direction'])
for mo in range(1, 7):
    mostr = f'2021-{mo:02}'
    df = pd.merge(factor.loc[mostr], returns.loc[mostr])
    x = df['roe']
    x = sm.add_constant(x)
    y = df['pct_chg']
    est = sm.OLS(y,x).fit()
    significant = (est.pvalues['roe'] < 0.05)
    direction = np.sign(est.params['roe'])
    result.loc[mostr] = significant, direction
```

4. 李明，某量化基金经理，正在测试逻辑回归模型的预测能力。他认为，根据个股的市盈

率和市值大小可以判断出个股是否存在投资机会。他将个股市盈率和对数市值作为逻辑回归模型的特征数据储存在文件 `data_x.csv` 中，将个股当年的涨幅数据储存在文件 `data_y.csv` 中。

`data_x` 部分数据如下：

code	pe	log_mv
600000.SH	4.6834	17.1949
600004.SH	-77.0214	14.7886
600006.SH	16.9126	14.1939
600007.SH	21.0572	14.4083
600008.SH	13.382	14.6083

`data_y` 部分数据如下：

code	change
600000.SH	0.0331
600004.SH	-0.2088
600006.SH	-0.2267
600007.SH	0.4726
600008.SH	0.0888

现需要你根据以下步骤描述，编写相应代码。

**4.1.** 定义函数 `standardize` 用于数据标准化。假设函数参数为 `data_x` 中的一列特征数据 `x`，函数返回值为标准化之后的特征数据，标准化公式为：

$$x^* = \frac{x - \bar{x}}{\sigma}$$

使用上述函数对特征数据进行标准化，并将标准化后的数据保存在变量 `data_2` 中。（4 分）

**参考答案：**

**# 定义函数（2 分）**

```
def standardize(x):
```

```
    return (x - x.mean())/x.std()
```

**# 读取数据**

```
data_x = pd.read_csv('data_x.csv', index_col=0)
```

# 标准化（2分）

```
data_2 = data_x.apply(standardize)
```

4.2. 将个股当年涨幅数据 `data_y` 转换成是否上涨的标签数据并保存在变量 `data_3` 中。  
（2分）

参考答案：

```
# 读取数据
```

```
data_y = pd.read_csv('data_y.csv', index_col=0)
```

```
# 处理标签
```

```
data_3 = data_y.apply(np.sign)
```

4.3. 将特征数据和标签数据进行分组，其中 80%作为训练集，20%为测试集，分别将训练集和测试集的特征数据和标签数据存储于变量 `train_x`, `train_y`, `test_x`, `test_y` 中。（4分）

参考答案：

```
from sklearn.model_selection import train_test_split
```

```
train_x, test_x, train_y, test_y = train_test_split(data_2,  
data_3.values.flatten(), test_size=0.2)
```

4.4. 使用决策树模型进行模型训练。（4分）

参考答案：

```
from sklearn.tree import DecisionTreeClassifier
```

```
clf_DT = DecisionTreeClassifier()
```

```
clf_DT.fit(train_x, train_y)
```

4.5. 使用测试集数据对模型进行测试，根据测试集特征预测标签，并将预测的结果保存在变量 `predict_y` 中。（2分）

参考答案：

```
predict_y = clf_DT.predict(test_x)
```

4.6. 根据以下公式计算模型准确率：

模型准确率=测试集中预测正确的次数/总的预测次数

(4 分)

参考答案：

```
accuracy = (test_y == predict_y).sum()/len(test_y)
```

5. 在投资过程中，风险管理对于稳定投资收益具有重要作用。在险价值 VaR 衡量了在在一定置信水平下，某资产或资产组合在未来特定一段时间内的最大可能损失。李明，某量化基金经理，希望使用在险价值 VaR 来衡量某投资标的的投资风险，他已将该投资标的的近一年的价格数据保存于文件 data.csv 中，部分数据如下：

date	open	close	high	low
2020-01-02	22.68	22.88	23.18	22.5
2020-01-03	22.89	23.04	23.2	22.71
2020-01-06	22.94	23.23	23.33	22.76
2020-01-07	23.23	23.15	23.28	22.91
2020-01-08	23.01	22.64	23.03	22.6

李明想观察该投资标的收盘价的收益率分布是否符合正态分布。请你：

5.1. 计算该投资标的收盘价的日收益率序列保存于变量 returns 中，数据类型 pandas.Series，使用算数收益率计算方法：(4 分)

参考答案：

# 导入模块

```
import numpy as np
```

```
import pandas as pd
```

```
from matplotlib import pyplot as plt
```

# 读取数据 (2 分)

```
data=pd.read_csv('data.csv',index_col=0,parse_dates=True)
```

# 计算算数收益率 (2 分)

```
returns=data['close'].pct_change().dropna()
```

5.2. 绘制 QQ 图，并判断该投资标的的日收益率是否符合正态分布；（4 分）

**参考答案：**

# 导入模块（1 分）

```
import matplotlib.pyplot as plt
```

```
from scipy import stats
```

# 绘制图形（2 分）

```
plt.hist(returns, bins=30, density=1)
```

```
mu = returns.mean()
```

```
sig = returns.std()
```

```
x = np.linspace(mu-3*sig, mu+3*sig)
```

```
y = stats.norm.pdf(x, mu, sig)
```

```
plt.plot(x,y,c="red")
```

# 解释图形（1 分）

该投资标的的日收益率分布相对于正态分布呈现尖峰肥尾的特征。

5.3. 历史模拟法是一种常用的非参数计算 VaR 的方法，采用历史数据直接估计资产价格变化的概率分布。请你使用 data.csv 文件中最近 120 天的数据，使用历史模拟法计算 95%和 99%置信度水平下的日在险价值（Daily VaR），分别存储于变量 dVaR\_95\_hs 和 dVaR\_99\_hs 中。（4 分）

**参考答案：**

# 最近 120 天收益率（1 分）

```
recent_rs = returns[-120:]
```

# 计算 95%的 Daliy VaR（2 分）

```
dVaR_95_hs = np.percentile(recent_rs, 5)
```

# 计算 99%的 Daliy VaR（1 分）

```
dVaR_99_hs = np.percentile(recent_rs, 1)
```

5.4. 当能够确定资产收益近似服从正态分布的情况下，可以使用参数法计算在险价值 VaR。请你使用 `data.csv` 文件中最近 120 天的数据，使用参数法计算 95%和 99%置信度水平下的日在险价值（Daily VaR），分别存储于变量 `dVaR_95_param` 和 `dVaR_99_param` 中。（4 分）

**参考答案：**

# 计算收益率均值和标准差（2 分）

```
mu = recent_rs.mean()
```

```
sigma = recent_rs.std()
```

# 计算在险价值（2 分）

```
dVaR_95_param = mu - stats.norm.ppf(0.95) * sigma
```

```
dVaR_99_param = mu - stats.norm.ppf(0.99) * sigma
```

5.5. 绘制该标的资产最近 100 天日收益率的频率分布直方图，并标注参数法下 95%VaR 和 99%VaR 所在位置。（4 分）

**参考答案：**

# 绘制频率分布直方图（2 分）

```
plt.figure()
```

```
plt.hist(recent_rs, bins=50, density=True)
```

# 标注 VaR 的位置（2 分）

```
plt.axvline(dVaR_99_param, color='r', label = '99% VaR')
```

```
plt.axvline(dVaR_95_param, color='y', label = '95% VaR')
```

```
plt.legend()
```

```
plt.xlabel('returns')
```