

PRESENTATION FOR NCSoft PROJECT

빅데이터 전문가 김민수

CONTENTS

01. Title 보유역량

02. Title 핵심성과

김민수

Data Analyst

Tech skills

Python

SQL

Excel

PySpark

Soft skills

Git

Confluence

PROJECT TITLE

투자 성과 극대화 투자 전략 기술 생성 및 개발

틱데이터 통계치 ETL 자동화

Module Name : tickdata_additional_stat_module

Class Name : Statistics

Purpose

: KOSPI 200, KOSDAQ 150 통계데이터 추출

Main Point

: 데이터 I/O 최소화

Workflow

1. 네이버, 카카오 증권 데이터 <-> 부서 내 데이터
일자 별 누적 통계치와 비교했을 때, 수치 차이 발생
-> 해당 원인 **트러블 슈팅 실시**

1-1 : RDBMS Table 파티셔닝 일자의 마지막 체결 / 호가 시간 로그 확인
-> 정규장 + 시간외 종가 (**시간외 단일가 데이터 미존재**)

1-2 : 같은 데이터 벤더 회사 프로그램을 사용하는지 확인
-> 동일 회사 프로그램 사용 확인 (**에프엔가이드 사**)

1-3 : 최종적으로 네이버 증권 사 문의를 통해 누적 통계치 집계 기준
확인 및 비교
-> **집계치 기준 차이 식별**

09.04.10 검수 결과 및 통계 데이터 추출

작성자 : 이상혁(Lee Sanghyeok)(sanghyeoklee), 최근 변경 : 김민수(Kim Minsoo)(minsu5123) 1분 이내

문의

- 구동 및 코드문의: @김민수(Kim Minsoo)(minsu5123)
- 업무논의: @박희환(Park Heehwan)(heehwanpark)
- @이상혁(Lee Sanghyeok)(sanghyeoklee)

검수 및 데이터 추출 개요

- factor inspection module : 팩터 검수 자동화 모듈 (전체 데이터 보유 여부 점검 / 기존 팩터 도출 여부 점검 / 모델 기반 팩터 도출 여부)
- tickdata_inspection_module : 틱데이터 전수 검수 자동화 모듈
- tickdata_additional_stat_module : 전체 RIC의 일자 별 통계데이터 추출/가공/적재 자동화 모듈

사용자가이드

factor_inspection_module

tickdata_inspection_module

tickdata_additional_stat_module

- # stocky_test.refinitiv_tickhistory_rics_statistics
- <Main Module : tickdata_additional_stat_module>
- <Module 구동>

stocky_test.refinitiv_tickhistory_rics_statistics

<Main Module : tickdata_additional_stat_module>

method	용도	output
execute	모듈 실행	
extract_tickdata	<refinitiv.kr_stock_tick_time_sales> ric,일자, 거래량,거래대금 추출	pd.dataframe
stat_preprocess	KML00 적재 전 DB Columns,Type에 맞게 dictionary 변환	
insert_result	KML00 stocky_test.refinitiv_tickhistory_rics_statistics 관련 결과물 적재	
_make_insert_query	insert 쿼리 작성	str

<Module 구동>

- 통계치 추출 일자 입력
- extract_query method 실행 → 거래량,거래대금 추출
- data_preprocess method 실행 → 적재 전 데이터 전처리
- execute method 실행 → KML00 결과물 적재

팩터 검수 자동화

Module Name : factor_inspection_module

Class Name : InspectionModule

Purpose

: 종목 별 팩터 보유 여부 검수

Workflow

1. 기존 프로토타입 모듈 로직 전면 수정 및 모듈 동작 여부 점검
(지속적인 CI (bitbucket) -> local 동작 완료 확인)
2. Apache Zeppelin Pyspark 환경에서 동작 할 수 있도록 코드 리팩토링(2회)
-> 시간적 런타임 효율성 약 60배 개선
3. 분석 파이프라인 확장 필요성 점검 및 추가 설계
(Extract -> Transform / Load 로직 추가)
4. 프레임워크 변경 시 발생하는 SQL Query 결과 미반영 오류 해결

페이지 / 금융기술개발실 홈 / 09. 공용 업무 / 09.04 Tick History & Factor 데이터 검수

09.04.10 검수 결과 및 통계 데이터 추출

작성자 : 이상혁(Lee Sanghyeok)(sanghyeoklee), 최근 변경 : 김민수(Kim Minsoo)(minsu5123) 1분 이내

문의

- 구동 및 코드문의: @김민수(Kim Minsoo)(minsu5123)
- 업무논의: @박희환(Park Heehwan)(heehwanpark)
- @이상혁(Lee Sanghyeok)(sanghyeoklee)

검수 및 데이터 추출 개요

- factor_inspection_module : 팩터 검수 자동화 모듈 (전체 데이터 보유 여부 점검 / 기존 팩터 도출 여부 점검 / 모델 기반 팩터 도출 여부)
- tickdata_inspection_module : 틱데이터 전수 검수 자동화 모듈
- tickdata_additional_stat_module : 전체 RIC의 일자 별 통계데이터 추출/가공/적재 자동화 모듈

사용자가이드

factor_inspection_module

tickdata_inspection_module

tickdata_additional_stat_module

- #stocky_test.refinitiv_tickhistory_mbf_inspection
 - <Main Module : InspectionModule CLASS>
 - < Module 구동>

#stocky_test.refinitiv_tickhistory_mbf_inspection

전체 총 검수 RIC : 493 / 499

제외 RIC list-up : 217600.KQ , 097520.KQ , 041140.KQ , 290510.KQ , 108320.KQ , 178920.KQ

<Main Module : InspectionModule CLASS>

method 명	용도
execute	전체 method 실행
extract_tns_trade	type="Trade"일 경우 필수적으로 보유해야 할 각 RIC의 refinitiv_time_n_sales columns value 보유 여부 : 인
extract_tns_quote	type="Quote"일 경우 필수적으로 보유해야 할 각 RIC의 refinitiv_time_n_sales columns value 보유 여부 : 확인
extract_md	market_depth 테이블의 각 컬럼 null 값 존재 여부 확인 → factor_check 구현 활용
extract_closed_market_issue	장중 bid,ask 호가 null 값 보유 ric 확인 → factor_check new columns add-in
inspection_process	전체 ric의 normal,model-based factor 추출
insert_result	전체 ric의 해당 일자 factor 검수 결과 적재 (stocky_test.refinitiv_tickhistory_mbf_inspection)
_preprocessing	전체 ric의 factor 검수 결과 dict 변환
_make_insert_query	kb_common 활용, insert query 생성

틱데이터 이슈 전수검사

Module Name : tickdata_inspection_module

Class Name : TickdataInspection

Main Point

데이터 I/O 최소화

Workflow

1. 메타데이터 정의 및 DB 구축,설계

1-1. Each Stocks Issue

(거래정지 / 기존 팩터 도출 오류 / 모델 기반 재정의 팩터 도출 오류)

1-2. Issue 세분화

: 현물 주식 거래정지 사유 + 팩터 오류 세분화

2. 틱데이터 전수 검사 모듈 코드 설계

목표 : 검수일자 입력 시 해당 일자 전체 종목에 대해 전수검사 실시

2-1 서로 다른 DB에서 고유 종목 ,각각의 Factor 값 추출

-> factor 보유 여부 비교 후 Issue 할당

2-2 거래정지 / 팩터 오류(normal / model_based) 로직 구현

2-3 거래재개 로직 추가 구현

2-4 결측,이상치, DB 형식에 맞게 전처리/적재 로직 구현

tickdata_inspection_module

- # stocky_test.refinitiv_tickhistory_inspection_result
- <Main Module : tickdata_inspection_module>
- <Module 구동>

stocky_test.refinitiv_tickhistory_inspection_result
<Main Module : tickdata_inspection_module>

method	용도	output
execute	모듈 전체 실행	x
extract_issue_info	<stocky_test.refinitiv_tickhistory_inspection_issue_info> 전체 issue 목록 추출	dict
extract_factor_info	<stocky_test.refinitiv_factor_inspection_result> 특정 factor 추출	x
extract_ric_info	refinitiv_time_n_sales 해당 파티션 일자의 ric 추출	x
inspection_process	해당 일자 issue RIC 자동 Issue Mapping	dict
insert_result	KML00 stocky_test.refinitiv_tickhistory_insepection_result 작업 결과물 적재	x
_make_insert_query	insert 쿼리 생성	str

<Module 구동>

```
%pyspark
daily_issue = TickdataInspection(date='20230329')
daily_issue.execute()
```

Took 3 sec. Last updated by anonymous at June 13 2023, 4:32:26 PM.

1. 틱데이터 전수검사 일자 입력

2. execute 실행

1. Technical indicator 기술적 지표 전략 구축 (총 34개)
2. Technical indicator 수치 추출 모듈 개발(Python)
3. 로직 백테스팅 및 관련 지표 데이터베이스 구축
3. Bloomberg 활용 글로벌 자산군 분류 및 기술
> 크로스 에셋 전략 및 향후 리스크 프리미엄 분석 활용

Four Category of Technical Indicator

작성자 : 이주영(Lee Juyeong)(jlee94), 최근 변경 : 김민수(Kim Minsoo)(minsu5123) - 2023-01-19

개요

- 목적
 - 가격, 거래량의 패턴을 설명하는 각종 지표 정리
 - 딥러닝 모델의 입력 변수 및 종목 설명의 근거로 활용
- 구성
 - Trend(추세) : 추세의 진행 방향,강도 분석하여 수익 실현
 - Momentum(모멘텀) : 추세의 가속도,강세를 분석하여 수익 실현
 - Volatility(변동성) : 현재 가격의 방향이 아닌 가격의 변동성을 이용하여 수익 실현
 - Force(거래량) : 거래량을 이용하여 현재의 추세 및 변동성의 강도를 확인하고 수익 실현
- 내용
 - Argumentation(전달인자) > Quantiwise Code 변환 , 각 Indicator 최신 Sample 추출 , Trend Category> SMA(단순이동평균) 추가
 - 각 지표 별 값을 도출하기 위한 일자 : Sample Columns에 날짜 표기 (없을 시 권장 Default 값 사용)
 - 하나의 Technical Indicator가 자료마다 Formula가 상이하여 , 최대한 원어 논문을 참고하여 수정(없을 시 대체 자료 인용)
 - 대체 자료 example : <https://www.fidelity.com/learning-center/trading-investing/technical-analysis/technical-indicator-guide/volume-oscillator>
 - Lookup Columns , Function Columns 추가 (23.01.09 일자)
 - Lookup Columns : 각 지표 별 분석 기법의 lookup 기간 선정
 - Function Columns : Calculation Sheet에서 Excel 함수로 산출된 값
 - Calculation Sheet 추가(*Function Columns 값 이해 자료)
 - (2022.01.04 ~ 2023.01.04) 삼성전자(A005930) 1년 Data 추출 >Excel 함수식 이용하여 값 도출 -> wiki에 파일 업로드(23.01.06) → 향후 검토 및 수정 후 최종본 업로드 예정
 - 카테고리 별 Technical indicator 분석 기법 추가 업데이트 (23.01.13 일자)
 - 기술적 분석에 자주 활용되는 indicator extra list-up (Channel Commodity Index , Elder Ray Power, Annual Percent Change, Donchian, StochRsi, Ease of Movement, A/D)
 - Reference > <https://www.elearnmarkets.com/blog/best-25-technical-indicators/>
- 참고
 - Sample Columns > ~23.01.04일 기준 Sample 추출 = 새롭게 추가된 Indicator 혹은 기존 Sample 계산 오류로 인한 수정

Thank you