**Förutspå volatiliteten för stockholmsbörsens index OMX.**

steg 1:

Ladda ner data från yfinance(yahoo)

Steg 2:

Gör en ML modell och mät nogrannhet.

Bygg eventuellt en testdel för backtestning. Andra tidsserier etc.

Förbättra modellens noggrannhet.

**Egna tankar:**

Ta en längre tidsserie, ta flera mindre tidsserier kortare perioder.

Datan har seasonal patterns(även om subtila) och Linjär tillväxt.

Problemdefinition:

Hitta en parameter som skulle kunna ge en riskindikation för ett finansiellt instrument.

Att skapa en indikation om framtida riktning av ett finansiellt instument, egentligen om det kommer att stå högre eller lägre inom nära framtid. För att kunna ha tillsammans med övriga indikatorer såväl makroekonomiska fundamentala som volym och kurs-tid baserade tekniska indikatorer, för att delvis (eller helt vid automatisk handel) ta bort människans psykologiska problematik vid handel. Indikationen skulle också kunna tjäna som en del i ett beslutfattningsstöd för en organisation.

Typen av problem blir någon typ av Regression, där man förutspår framtida värden med hjälp av tidsseriedata.

**Data:**

Trender i datat har inga tydliga mönster, men man kan se trender över olika tid, inom en dag, inom flera dagar, och mer sesongsbetonat, även om det är diffust. SARIMA skulle kunna användas, men jag har testat detta och det ger inte en tydlig bild av marknaden, då skulle man behövas andra funktioner för att hantera speciella händelser för att försäkra sig mot större risker.

Outliers existerar, likväl som att kursen rasar/peakar på kort sikt och orsakar problem för trenden.

Outliers kan man främst se i dem, de man eventuellt skulle kunna sortera bort är när det är låg volym, t ex under natt eller när det är helgdagar under stora delen av världen eller länder som genererar hög volym i marknaden.

Man skulle också kunna sortera bort ras och extrema värden vid nyheter, vilka ofta återgår till kursen innan nyheten och har en varaktighet på under 15 minuter. Dessa extrema svängningar skulle kunna hanteras av sk stopploss system som automatiskt går ur marknaden vid en viss rörelse ifrån köpkurs, eller volatil rörelse över tid(dynamisk stoploss). Om man har data för minut och längre tid kan man lättare behandla outliers eller kortsiktiga extrema svängningar som är vanligast vid nyhetsinformation. I realtid skulle det kunna uppstå problem med Pythons lägre hastighet jämfört med ex vis C++. Det går endast att få 1 min för de senaste 7 dagarna med Yfinance, det blir att betala för att få bra minut data över lång tid. Dessutom kräver det en hel del processorkapacitet, varvid detta undvikes för denna typ av initiala analys.

**Gratis finansiell data:**

● YFinance: Flera typer av finansiell data.

● FRED: Ekonomi och marknads data

○ En stor mängd ekonomisk och finansiell/fundamental data

● Alpha Vantage: Equities, valutor, crypto, sektorer, indikatorer.

○ RealtRealtids och historisk API

○ Ostabila endpoints och data reader behöver en fix

● Quandl: One stop shop för alla typer av data

○ Free and premium APIs

● IEX: Equity, ETF market and fundamental data

● Polygon.io Låg latency data i fh till YFinance.

○ Gratis och premium APIs

● EDGAR: Företagsfundamentala data

● Hedged Capital: ML generade marknads analyser.

○ Insights about which factors moved the markets in the prior week

● Finnhub API

○ Har pre-market data, som inte t ex Yfinance har. Kräver nyckel.

YFinance:

---------

Finns i följande upplösning:

1m, 2m, 5m, 15m, 30m, 60m, 90m, 1h, 1d, 5d, 1wk, 1mo, 3mo

Möjliga datavariabler för en aktie(apple):

För individuella kolumner:

aapl\_historical[“High”]

aapl\_historical[“Low”]

exempel:

aapl\_historical = aapl.history(period="max", interval="1wk")

aapl\_historical

Ger kolumner med veckodata över längre tid.

info() ger information om hög låg:

dayHigh

dayLow

fiftyTwoWeekHigh

fiftyTwoWeekLow

pyfin tickers:

^TNX 10 års ränta ränteobligationer, ränta på stadsobligationer

^FVX

^VIX

^IRX 13 veckors Treasure bill statsobligationer

^TLT iShares 20+ Year Treasury Bond ETF (TLT)

Yfinance har också en modul som heter ticker som har många attribut om man t ex vill visualisera olika typer av data för tickern i python. Yfinance är dock inte särskilt väl uppdaterat, en del funktioner har tagits bort och en del ändrat namn. Man kan få stora problem med datat om man gör för många eller för stora anrop för ofta. Däremot är det ett enkelt sätt att få realtidsdata ner till 1 minut utan att behöva en nyckel. Man bör ej använda detta till att handla med då det inte kan anses vara så pålitligt, men det är bra för analyser som inte kräver ständig uppdatering och snabbhet.

---------------------------------------------------

Logistisk regression är mycket snabb, men har få hyperparametrar.

XGBoost är långsam om man skall använda till mer realtidsliknande tradingapplikationer. Men har många hyperparametrar, kan bli komplicerad att hantera/förstå.

Stacking, ensamble learning skulle kunna förbättra resultatet.

Cross validation skulle kunna vara ett val.

Hyperparameter tuning skulle kunna ge svar om man har fler input, även fundamentala data, minska(sortera) volatilitet för att få fram sharp ratio mfl. Då skulle olika search för att få fram nya parameters, t ex Grid search, Random search, Bayesian optimization, etc.

ARIMA och sarima tar inte hänsyn till volatiliteten om den inte är konstant över tid. ARCH och GARCH har en bättre funktionalitet för detta. Volatilieten är mycket viktig för riskhanteringen, som är avgörande för avkastningen för den som handlar i marknaden, eller gör finansiella prognoser för företag och organisationer.

**Modeller som kan användas för finansiella data över tid:**

Arima står för AutoRegressive Integrated Moving Average. Den har 3 parametrar: p, d och q. Parametern d representerar graden av skillnad eller integration. q-parametern representerar antalet glidande medelvärden eller feltermer.

KERAS anses vara mycket bra för trading, när en deep learning modell skall implementeras.

\*\*arma och arima och sarima

dubbel exponentiell utjämning

witers metod.

\*\*LSTM med PYTORCH, KERAS, SKLearn

LSTMs are a type of Recurrent Neural Network (RNN)

ARIMA, SARIMA, ARCH, GARCH Kan mycket väl användas till denna typ av Timesserie data.

\*\*Facebook Prophet

Vector Autoregressive(VAR) används också för tidserier, för att fånga utvecklingen och förhållanden(korrelationer). Det är en ekonometrisk modell som involverar multipla tidsserier.

Övriga ai funktioner och bibliotek, för att förutspå trender och för att normalisera eller mjuka ut trendrörelser, samt för att simulera händelseförlopp:

Smoothing Constants

Alpha (level)

Gamma (trend)

Tidsserieregression, linjär trend

Holt’s metod

Monte Carlo simulation.

Facebooks prophet för aktier.

Random Forest

Lasso, Ridge Regression L1, L2 Regularization.

Markov chains när det går väl att använda de allra senaste tidpunkterna för att förutspå kommande nivå, den är alltså liknande xcboost om man använder någon form av smoothing till det, ex. vis Kalman filters.

**Plattformar för python programmering**

Alpaca, Tradier, Blueshift, pynance, även MT5, men framförallt Quantopian som är en gratisplattform med många funktioner för att bygga och exekvera strategier i python, Quantconnect är också en liknande plattform, där man kan programmera och visualisera enkelt med oliKA tekniska indikatorer.

Man kan också för att kunna implementera fler Python funktioner använda färdiga specifika financiella python package som tex ffn, vollib, pyfin,Quandl, FinancePy, Zipline, Pyfolio, Tidsserie:tia, dynts, Arch, mfl.

Resultatet blev en volatilitets prognosticering, avkastning av OMX dagligen, mha arch: garch, bootstrap:

3 horisonters prognos med Garch metod av arch:

En bild som visar text, skärmbild, Teckensnitt, diagram

Automatiskt genererad beskrivning

Garch med bootstrap:

En bild som visar text, skärmbild, Teckensnitt, diagram

Automatiskt genererad beskrivning

Man kan också för att kunna implementera fler Python funktioner använda färdiga specifika financiella python package som tex ffn, vollib, pyfin,Quandl, FinancePy, Zipline, Pyfolio, Tidsserie:tia, dynts, Arch, mfl.

Slutsats:

Med Arch och Garch Skapat h-step framåt, genererat för de 3 horisonterna.

Jag har använt metoden bootstrap för att mäta kvantiteter genom att skapa medelvärde av multipla flera samplingar. Optimerat genom att välja samplingar och repetitioner. Man kan ersätta naive estimering, GARCH modellen med bootstrap metoden funkar väl när det gäller förutspå volatiliteten för index. Arch modellen kan ge indikation om större rörelser i volatilitetskluster, men hittar intetillräckligt många samband i volatilitetsklustren. Detta visade sig tydligt med Lagrange Multiplier test (LM test

även benämns Engle’s Test for Autoregressive Conditional Heteroscedasticity.

Modellens alla ARCH effekter skall vara med. Då jag fick klart under 0.5 i värden anses det inte vara tillräckligt. Att använda garch efter arima eller sarima är inte lämpligt, då MA/medelv. funktioner gör att klustren och t ex högre volatilitet begränsas. Garch metoden hämtar ju många små punkter och hittar klusterdistributioner av värden över tid. Det går ändå att använda dem parallelt eller under olika cykler.