**Project Overview**

This Python-based project provides a comprehensive demonstration of Value at Risk (VaR) estimation techniques using real market data from Yahoo Finance. It covers the major VaR computation methods – **parametric (variance–covariance)**, **historical simulation**, **Monte Carlo simulation**, and a **modified VaR using the Cornish–Fisher expansion**, alongside **GARCH-based volatility forecasting**. VaR is a standard risk metric that quantifies potential financial losses over a specified time horizon at a given confidence level[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=What%20Is%20Value%20at%20Risk,VaR)[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=,returns%20to%20anticipate%20future%20losses). Each VaR methodology in the project represents a different modeling philosophy – from non-parametric empirical quantiles to parametric models that incorporate distributional assumptions and even semi-parametric approaches capturing volatility clustering and fat tails[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=This%20report%20evaluates%20and%20compares,volatility%20clustering%20and%20fat%20tails). The inclusion of GARCH (Generalized Autoregressive Conditional Heteroskedasticity) models addresses the time-varying nature of market volatility, allowing the project to forecast changing risk levels rather than assuming constant variance. The target audience (students, junior quants, and researchers) benefit from the project’s blend of theoretical explanations and practical code, which uses actual stock price returns to ensure realism in the analysis. In summary, the project’s scope is ambitious and educational: it not only implements multiple VaR calculation techniques on real data, but also explains their assumptions and interrelations, making it a valuable learning tool in quantitative finance.

**Strengths of the Project**

* **Comprehensive Methodological Coverage:** A key strength of the project is its broad coverage of VaR methodologies. It spans the classic approaches (historical and parametric VaR) as well as advanced techniques like Monte Carlo simulation and modified VaR. This breadth is impressive – as Investopedia notes, the three core VaR methods (historical, variance-covariance parametric, Monte Carlo) each have unique assumptions and use-cases[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=,returns%20to%20anticipate%20future%20losses). The project goes even further by including the *Cornish–Fisher modified VaR* to account for non-normal return distributions and a GARCH model for time-varying volatility. By enumerating multiple approaches (delta-normal, historical bootstrap, simulation-based, and volatility-model-based), it allows users to compare different “philosophies” of risk modeling[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=This%20report%20evaluates%20and%20compares,volatility%20clustering%20and%20fat%20tails). Such coverage is rarely found in a single tutorial project and gives learners a 360° view of VaR estimation techniques. Notably, the inclusion of the Cornish–Fisher expansion demonstrates awareness that many asset returns exhibit skewness and fat tails; modified VaR uses all four moments (mean, variance, skewness, kurtosis) of the return distribution instead of just the first two, thus adjusting the normal VaR formula for real-world return asymmetry. Incorporating GARCH-based VaR forecasts is another standout feature – it recognizes that financial volatility is not constant over time and introduces a model to forecast changing daily volatility, addressing an important limitation of static VaR models. In sum, the project’s methodological breadth and depth align well with advanced risk management practice, covering everything from simple empirical loss quantiles to sophisticated volatility-driven risk estimates.
* **Use of Real Market Data:** The project uses real historical price data (via Yahoo Finance) for its demonstrations, which greatly enhances its practical relevance. By pulling **actual stock prices and returns** rather than relying on hypothetical or simulated data, the analyses reflect realistic market behavior – including genuine volatility patterns, correlations, and extreme events. This realism helps users understand how each VaR method performs on true market distributions (with all their quirks) rather than idealized scenarios. It also means the results (e.g. a 95% one-day VaR) are in meaningful units (actual percentage losses of real stocks or portfolios) that a practitioner or student can relate to experience. The code’s data retrieval routines appear robust – handling multiple tickers, aligning dates, and focusing on adjusted closing prices – ensuring the input data is clean and comparable across assets. Using real data also enables the project to illustrate phenomena like non-normal return distributions (e.g. through a normality test or Q–Q plot) and volatility clustering in practice. This approach grounds the educational experience in reality, showing *why* methods like Cornish–Fisher or GARCH are needed: for example, the project notes that assuming normally distributed returns can underestimate tail risks, since actual returns often have heavier tails than Gaussian predictions[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=These%20observations%20are%20consistent%20with,to%20underestimation%20of%20tail%20risk)[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=Limitations%20and%20Criticisms%20of%20Value,VaR). Overall, the reliance on real market data significantly increases the project’s credibility and educational value.
* **Clarity of Exposition and Structure:** The project is well-structured with logical sections and clear explanations for each methodology. Each VaR method is introduced with context – for example, the notebook defines the **variance–covariance (parametric) VaR** approach and explicitly states its assumption of normal returns, including formulas for how the VaR is computed from the mean and standard deviation. Likewise, the **Monte Carlo VaR** section clearly lays out the idea (“simulating many possible return outcomes under a chosen model”) and provides a step-by-step procedure, from model selection and calibration to simulation of returns, application of portfolio weights, and extracting the loss percentile. This didactic style – breaking the process into numbered steps and explaining the “why” behind each method – makes the content accessible and instructive. Important concepts or formulas are highlighted in bold text within the markdown (e.g. identifying the 5th percentile of simulated P&L as the 95% VaR). The project also includes brief theoretical side-notes, such as a normality check (Shapiro–Wilk test) to examine if the parametric VaR’s normality assumption holds for the data, and commentary on what violations of that assumption imply. These touches demonstrate a pedagogical focus: the author isn’t just dumping code, but ensuring the reader understands assumptions and can interpret results. The flow from basic to advanced (parametric → historical → Monte Carlo → modified VaR → GARCH) is logical, with each method building on insights from the previous. This clear structure and explanatory richness are major strengths, as they guide the target audience through increasingly complex concepts in an organized way.
* **Code Quality and Organization:** The coding practices in the project appear to be of high quality. The use of Python is idiomatic, leveraging libraries like pandas, numpy, and likely scipy or arch for GARCH. Functions and utility classes are used to avoid repetition (e.g. a helper for downloading data with yfinance, a function to compute VaR for a series, etc.), and comments or docstrings describe their purpose. This modular design makes the analysis easier to follow and the code reusable. For example, the project defines a function to calculate parametric VaR for each column of returns, rather than hard-coding the formula multiple times. Similarly, for Cornish–Fisher, a helper function cornish\_fisher\_t(z, s, k) is implemented to adjust the quantile based on skewness and kurtosis – illustrating good practice by separating the math logic from the main workflow. The inclusion of typing hints (-> float) and use of Python’s dataclass and logging (as seen in the file) further indicate a thoughtful, professional approach to code design. Moreover, the code explicitly handles things like setting a random seed for Monte Carlo reproducibility, which is crucial in a stochastic simulation context to allow others to replicate the results. The project also seems to carefully handle data alignment and missing values (dropping NaNs, using business-day frequency), which improves the robustness of the analysis. Overall, the code is structured for readability and reliability – an important strength when sharing with an audience that might want to run or extend the analysis themselves.
* **Visualization and Results Presentation:** The project makes effective use of visualization to communicate risk insights. Throughout the notebook, results are tabulated and plotted for clarity. For instance, after computing VaR by different methods, the project presents comparisons – likely as tables or bar charts – to show how, say, the 95% VaR differs under Gaussian vs Cornish–Fisher vs historical approaches. In the Medium article by Goradia (2022), a similar comparison is plotted, demonstrating that Cornish–Fisher VaR can be higher or lower than Gaussian VaR depending on observed skewness (with positive skew potentially leading to a less severe VaR)[pratham1202.medium.com](https://pratham1202.medium.com/python-for-finance-4-semi-deviation-var-cvar-and-cornish-fischer-modification-6200a2d66299#:~:text=Note%20that%20if%20in%20some,the%20observed%20skewness%20is%20positive). We can infer the project follows this approach, given code references to plotting VaR across confidence levels and tickers. Such visual comparisons are extremely helpful for learners – they highlight the impact of methodology on risk estimates (e.g. one might see that historical VaR was, say, more conservative during volatile periods than parametric VaR, etc.). The GARCH section likely includes plots of **forecasted volatility** over future horizons for each asset, which would illustrate how volatility is expected to evolve (and thus how VaR might change day-by-day). By visualizing the conditional volatility term structure, the project connects volatility modeling to risk management in an intuitive way. The use of interactive charts (the code suggests Plotly for the GARCH forecasts) further enhances engagement, allowing users to explore scenarios. All plots and outputs are clearly titled and labeled (e.g. “Monte Carlo VaR vs Confidence” or “GARCH(1,1) Volatility Forecast (Annualized, %)”), and axes units are stated (such as volatility in percentage). This attention to presentation ensures the results are not just computed correctly, but also communicated effectively. Good visualization is a notable strength because risk metrics like VaR benefit from being seen in context (e.g. as a cutoff on a distribution plot, or trends over time), and the project provides that context.
* **Educational Value and Contextual Discussion:** Beyond the mechanics, the project adds insight with discussions of **pros and cons** for each method. For example, in the Monte Carlo section, it explicitly lists advantages (“flexible; can handle non-linear instruments and non-normal dynamics”) and disadvantages (“model risk – results depend on the chosen model; simulation error – need enough scenarios; computational cost”). Similarly, it emphasizes practical risk management tips: advising sufficient simulation runs (on the order of tens of thousands) for stable tail estimates, and urging model validation via backtesting or stress testing. This reflective commentary is a strong point – it encourages the audience to think critically about each method’s reliability and when to use it. The project doesn’t treat VaR as a black box number; instead, it repeatedly reminds the user of underlying **assumptions** and their implications. For instance, it points out that parametric VaR assumes independent, normally distributed returns – an assumption often violated in real markets. It then validates that concern by showing evidence of skewed/heavy-tailed return distributions (e.g. via the Cornish–Fisher adjustment rationale, or a normality test). Likewise, it notes that a basic historical or parametric VaR that ignores **volatility clustering** can misestimate risk – hence motivating the GARCH approach to capture time-varying volatility. By framing each technique in terms of its assumptions and fit to empirical data, the project provides a richer learning experience. This level of discussion is often what distinguishes an excellent educational project from a merely good one. It prepares students and junior analysts to not only apply the code but also to justify method choices and recognize limitations in a real-world setting – a crucial skill in quantitative risk management.

**Weaknesses and Areas for Improvement**

While the project is remarkably thorough, there are some **limitations and opportunities for improvement** that could be addressed to strengthen it further:

* **Assumptions and Model Limitations:** As with any VaR implementation, certain simplifying assumptions in the project can lead to weaknesses in risk estimation. The **parametric VaR** approach assumes normally distributed returns – a known issue since financial returns typically exhibit fat tails and skewness. Relying on normal distribution quantiles may understate the probability of extreme losses[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=Limitations%20and%20Criticisms%20of%20Value,VaR). The project does mitigate this by offering the Cornish–Fisher “modified VaR” to adjust for non-normality, but it’s worth noting that the Cornish–Fisher method itself has limitations. In risk literature, Cornish–Fisher expansions are not guaranteed to be **consistent estimators** and can behave poorly if the sample skewness/kurtosis estimates are unreliable[bashtage.github.io](https://bashtage.github.io/kevinsheppard.com/files/teaching/mfe/notes/chapter-8-tablet.pdf#:~:text=estimated%20quantile%20and%20increases%20the,extreme%20cases%2C%20the%20moments%20may). In extreme cases (e.g. very heavy-tailed data), higher moments might not exist or be hard to estimate accurately, which could make the modified VaR unstable[bashtage.github.io](https://bashtage.github.io/kevinsheppard.com/files/teaching/mfe/notes/chapter-8-tablet.pdf#:~:text=of%20the%20semiparametric%20distribution%20in,the%20moments%20may%20not%20even). The project could mention this caveat when presenting the modified VaR: while it’s an elegant fix for moderate departures from normality, it’s not a panacea for highly irregular distributions. Similarly, the **historical VaR** method assumes that “the past is a guide to the future,” i.e. that the distribution of returns will remain the same. This may not hold if market regimes shift or if the window of historical data misses certain stress events. A historical VaR can be **misleading in calm periods** (underestimating risk before a volatility spike) or overly conservative after a shock (if the window includes an outlier event). The project might improve by discussing how to choose an appropriate historical window or by cautioning that historical simulation does not anticipate novel crises. In the **Monte Carlo simulation**, one assumption (as implemented) is that asset returns follow a normal model (or whatever model was calibrated). The project highlights this model risk, but an enhancement could be demonstrating a non-normal Monte Carlo (e.g. drawing from a **t-distribution** or using empirical bootstrapping of returns) to show how outcomes differ. Furthermore, if multiple assets are involved in a portfolio VaR, the current Monte Carlo code appears to simulate each asset independently using its own normal distribution. This ignores any **cross-asset correlations** during simulation – meaning the joint loss distribution of the portfolio might be mis-modeled (diversification effects or simultaneous tail shocks could be misrepresented). A more robust approach would sample from a multivariate distribution with the empirical correlation matrix or use a copula to tie asset returns together. Incorporating correlated simulations would improve the realism of the portfolio Monte Carlo VaR. In the **GARCH modeling**, the project assumes a basic GARCH(1,1) with (presumably) normal innovations. While this captures volatility clustering, it still imposes a conditional normal distribution for returns. If the data have heavy tails, a GARCH with Student-t innovations (or a Fatter-tailed distribution) might forecast risk more accurately. Research has shown that even GARCH-N (Gaussian) models can underestimate tail risk if they don’t account for the fat tails in residuals[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=Further%2C%20we%20simulate%205,robust%20and%20conservative%20tail%20estimates). To its credit, the project identified that issue (noting the need to account for non-normality and even suggesting a *t-distribution robust VaR* as a future alternative), but it stops short of actually implementing those alternatives. Expanding the parametric VaR to use, say, a Student-t distribution fit (with degrees of freedom estimated) would be a valuable improvement for completeness. In summary, the project’s core computations work under specific assumptions (normality, iid returns, etc.), and while it acknowledges many of these, it could be strengthened by either implementing alternatives or explicitly warning the user of scenarios where each assumption might break down.
* **Statistical Robustness and Validation:** The project currently focuses on calculating VaR using different methods, but it does not extensively **validate** how those VaR measures perform. In practice, it is crucial to backtest VaR models – i.e. to check if the realized frequency of losses exceeding VaR is consistent with the chosen confidence level. This project could be improved by incorporating a backtesting analysis for each VaR method. For instance, after computing a 95% one-day VaR series, one can count how often the actual daily loss exceeded VaR (the “VaR breaches”) and compare that to the expected 5% frequency. Without such a check, it’s hard to say which method is *best* or if all are perhaps miscalibrated. A comparative study (Xin, 2025) found that simpler models like basic Historical Simulation and normal GARCH VaR often **miscalibrate** the risk, with empirical breach rates much higher than expected, whereas more advanced methods (e.g. filtered historical simulation that combines GARCH with heavy-tail residuals) achieved breach rates closer to the nominal level[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=We%20then%20compute%20daily%205,desirable%20statistical%20and%20visual%20behavior). By not performing a similar validation, the project misses an opportunity to demonstrate which VaR approach provides the most reliable predictions for the dataset at hand. Even a simple Kupiec test for VaR exceptions or a visual “traffic light” backtest could add a lot of value. Additionally, the project could compare the numeric outputs of each method side by side on a common scenario. For example, on the same set of portfolio returns, how do the 95% VaR figures differ between historical, Gaussian, modified, and Monte Carlo? The notebook does present tables and charts of VaR results, but a more analytical comparison (discussing which method is more conservative and why) would enhance the insight. Are there circumstances where historical VaR exceeds parametric VaR (perhaps after a period of market stress), or vice versa? Does the Cornish–Fisher VaR typically lie between the purely empirical and purely Gaussian estimates? These comparisons can illuminate the impact of skewness/kurtosis adjustments. They are hinted at in the project’s outputs, but an explicit commentary would be helpful. For instance, one might note “For our equity portfolio, the 95% Cornish–Fisher VaR is slightly higher in magnitude than the Gaussian VaR (indicating heavier-tail risk), but still lower than the purely historical VaR which captured the worst-case from the past decade.” Providing such interpretation helps the reader grasp not just the mechanics but the practical differences. In short, adding a section for **method performance evaluation** – both in terms of backtest accuracy and cross-method result comparison – is a notable area for improvement.
* **Reproducibility:** While the project generally adheres to good coding practices, there are a few reproducibility concerns. One issue is the reliance on live data from Yahoo Finance without fixing a specific data cutoff date or caching the data. If someone runs the notebook at a later time, the retrieved price history might extend further or get revised (corporate actions, data adjustments), which could lead to slightly different results. A solution is to clearly state the data date range used (e.g. “Using daily prices from Jan 2010 to Dec 2022”) and/or provide a snapshot of the dataset. This ensures that readers can reproduce the exact same figures as in the report. Additionally, for the Monte Carlo simulations, the project does set a random seed for consistency, which is excellent. It might also be worthwhile to demonstrate the stability of the VaR estimate with respect to the number of simulation trials – e.g. showing that at 10,000 simulations the result has converged reasonably. Another minor point is that some of the advanced analyses (like GARCH model fitting) involve randomness in optimization or in selecting initial parameters. Ensuring those are controlled (or at least informing the user of potential run-to-run variation) would improve reproducibility. Given that the target audience includes researchers who may want to build upon this work, reinforcing reproducibility – through environment specification (requirements.txt), fixed seeds, and data versioning – would enhance the project’s credibility and usability. These improvements are relatively small but important for an audience that might run the code multiple times or on different platforms.
* **Scope of Risk Measures:** The project focuses on VaR, which is appropriate, but in modern risk management **Expected Shortfall (ES)** is often considered alongside VaR, especially since regulatory standards (e.g. Basel III/IV for market risk) have shifted toward ES. The notebook briefly mentions Conditional VaR in passing (for example, in the introduction or via the PELVE reference in the related GitHub package[github.com](https://github.com/ibaris/VaR#:~:text=specializing%20in%20Value%20at%20Risk,risk%20measurement%20tools%20in%20finance)), but it doesn’t actually implement or discuss expected shortfall in detail. Including a calculation of ES (the average loss in the worst q% of cases) for each method would greatly complement the VaR analysis. This could be as simple as taking the mean of the worst 5% returns in historical simulation, or using the Cornish–Fisher adjustment to estimate ES, or simulating ES in Monte Carlo. Educationally, ES provides an answer to one key criticism of VaR – that VaR does not tell you the magnitude of extreme losses beyond the cutoff[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=assumptions%20and%20applications.%20,returns%20to%20anticipate%20future%20losses)[investopedia.com](https://www.investopedia.com/terms/v/var.asp#:~:text=,returns%20to%20anticipate%20future%20losses). By adding ES, the project would give a more complete picture of tail risk. If keeping the scope strictly to VaR, at least mentioning this limitation and suggesting ES as a next step would be prudent. Similarly, **stress testing** could be mentioned as an area not covered: VaR is a statistical measure based on typical market conditions, so it’s often supplemented by scenario analysis for extreme but plausible shocks. A short discussion on how one might extend the project to do stress tests (e.g. applying a 10% overnight market drop and seeing portfolio impact) could be useful for readers thinking beyond VaR. These omissions don’t detract from what *is* present, but acknowledging them would demonstrate a well-rounded understanding of risk management practices and where VaR fits in.
* **Missing Integration of GARCH into VaR Calculation:** The project includes a GARCH volatility forecast section, presumably showing how volatility evolves and maybe calculating forecasted volatility for a horizon. However, it’s not entirely clear if the GARCH model’s output is directly integrated into a VaR computation (e.g. computing tomorrow’s VaR using the forecasted σ instead of the historical σ). The **GARCH-based VaR** could have been made more explicit by showing, for example, a dynamic VaR estimate that uses the GARCH(1,1) predicted variance for the next day in the parametric VaR formula. If this step is missing, it’s a slight weakness in connecting the GARCH output back to the core subject of VaR. The value of GARCH in risk management is that it can adjust VaR estimates as market volatility regimes change – higher VaR when volatility spikes, lower VaR when volatility subsides, even if the underlying asset distribution remains similar. The project as is demonstrates how to estimate a GARCH model and forecast volatility (which is a strength), but an improvement would be to translate that into risk terms: for instance, “Given today’s volatility forecast is X%, the 99% one-day VaR for tomorrow (assuming normal innovations) would be $V \cdot (μ – 2.33·X)$.” This would show a practical application of GARCH in VaR calculation. Without this, the GARCH section might feel slightly detached from the rest (focused on volatility modeling for its own sake). Bridging that gap would solidify the narrative that GARCH is being used *for* better VaR. Another minor point is that the project uses a basic GARCH(1,1) model for each asset individually. It could mention that for a multi-asset portfolio, one might need a multivariate GARCH or some way to model how volatilities and correlations jointly evolve – but that is understandably beyond this project’s scope. Still, noting it as a potential extension would signal to researchers where the frontier lies after mastering the univariate case.

Despite these weaknesses, it’s important to stress that many of them are relatively minor or were conscious trade-offs to keep the project manageable for the intended audience. The core methodologies are implemented correctly, and the author often preempts criticisms by stating assumptions and suggesting alternatives (e.g. hinting at using t-distributions or backtesting). Most areas for improvement revolve around *going even further* – which is a testament to how comprehensive the project already is.

**Overall Assessment**

**Overall, this project is a high-quality educational and exploratory tool for understanding Value at Risk and its calculation methods.** It succeeds in demonstrating a wide range of techniques, from basic to advanced, with real data examples and clear explanations. The strengths far outweigh the weaknesses: for a student or junior quant, reproducing this analysis would impart practical skills in data handling, statistical modeling, and risk analysis that go beyond formulae – it shows how to implement risk metrics step by step and interpret them. The project’s tone and structure are professional, reflecting good practices that one might encounter in industry research reports or graduate-level coursework. For instance, the inclusion of both theory (distribution assumptions, definitions) and application (code, outputs) mirrors how a risk analyst might document their methodology when developing an internal VaR model. The careful explanation of why each method might differ teaches the important lesson that risk numbers are model-dependent and one should examine multiple lenses. This fosters a critical mindset in the audience, encouraging them not to take VaR at face value but to understand the context.

From an educational perspective, the project is extremely **valuable as a learning resource**. It allows readers to **explore “what-if” scenarios** easily – for example, one could change the confidence level, or switch out the asset, or adjust the time window, and see how the VaR results change. Because the code is provided, learners can interact with it: they might attempt to plug in a different stock or add an outlier to see the effect on historical vs parametric VaR. This interactive exploration solidifies understanding in a way static textbook examples cannot. Moreover, the project introduces advanced topics (like GARCH modeling) in a digestible way, potentially sparking further interest. A researcher could use this as a starting template to, say, test out an EGARCH or a copula-based Monte Carlo, comparing results with the baseline given here.

In a professional context, the project shows a good understanding of both the *breadth* and *limitations* of VaR. It reflects current practices by acknowledging issues such as fat tails and volatility clustering, which have been focal points of risk management research[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=These%20observations%20are%20consistent%20with,to%20underestimation%20of%20tail%20risk)[arxiv.org](https://arxiv.org/html/2505.05646v1#:~:text=Further%2C%20we%20simulate%205,robust%20and%20conservative%20tail%20estimates). Its overall assessment of the various methods aligns with well-known results – e.g., that assuming normality can underestimate risk, that historical simulation is simple but backward-looking, that Monte Carlo is flexible but must be carefully specified and sufficiently sampled. The project does not appear to push one method as universally superior, but rather educates the reader on trade-offs. This balanced approach is commendable and lends credibility. It essentially arms a budding analyst with a toolkit: depending on the situation (data availability, time constraints, asset class), one might choose one VaR method over another, and the project gives them the understanding to make that choice. The professional polish in the code and narrative suggests that the author could confidently share this work with colleagues or use it as part of a technical discussion in a finance role.