Risk Management Analytics Applied to Consumer Packaged Goods Companies

Pyxidr’s Value Proposition
CPG companies can use risk management techniques for pricing contracts and mitigating risks to seek higher risk/reward while meeting consumers’ needs.

Pyxdir has developed a flexible solution to easily model contractual terms (simple or sophisticated), report their positions in the underlying commodities, measure their financial risks and assess various pricing/hedging strategies in a user friendly environment.
2 KEY METRICS

Expected value and Value-at-Risk (“VaR”) are two metrics used for pricing contract features and measuring risk associated with a portfolio of contracts/positions.

**EXPECTED VALUE**

- A given contract structure leads to a distribution of values (e.g., gross margin) with an expected value – wider is the distribution, larger is the risk.
- A new feature (e.g., volume flexibility, cap/floor on prices) will lead to a different distribution and expected value. The difference in expected values represents the cost (premium to be charged) or the benefit (discount that could be given) associated with the feature.

**VALUE-AT-RISK**

- VaR is a statistical measure that gives the profit that could be lost (from the expected profit) for a given probability (5%) and time horizon.
- The value can be any financial measures (e.g., sale margin, EBITDA) and probabilities associated with some financial objectives (e.g., meeting budget) can be calculated.
INTERACTIVE DASHBOARD

We produce interactive dashboards to support commercial insights related to each contract type and explore alternative options or strategies.

Illustrate insights interactively – e.g.,
• Volume flex substantially impacts risk
• Customer rationality makes the impact asymmetric

Inputs: Interactive controls for varying relevant parameters:
• Allowed % variation
• If customers react rationally to prices (i.e., when commodity prices are low, other suppliers may be cheaper)

Outputs: Performance and risk metrics both with “baseline” and modified parameters:
• Expected margin, and margin at risk
• Visual distributions

1 It could be any contract types, simple or including complex features (e.g., volume flexibility, price caps/floors, etc.).
INTERACTIVE DASHBOARD

The same dashboards could be used to visualize the residual risk related to the portfolio of contracts and explore hedging strategies.

Show the aggregated position or margin based on:
- Contracts parameters, and
- Hedging program

Inputs: Percentage of the initial exposure hedged

Outputs: Performance and risk metrics both with “baseline” and modified parameters:
- Expected EBITDA, and EBITDA at risk
- Visual distributions
Pyxidr uses a 4-step approach to help their clients pricing various contractual terms and determining a coherent hedging program.

**MODEL**
- A model in Excel (or Google Sheets) for the cash flow in terms of the most important risk drivers
- It is based on what’s important to the business
- Can be very granular, splitting cash flows by contract
- Can represent very complex structures

**CALIBRATION**
- For each risk driver (e.g., commodity prices), calibrate a stochastic process to generate random values for it
- Can be based on historical values, or implied market price volatilities
- Can capture (auto/)correlation, mean reversion and volatility proportional to price

**RISK LEVEL**
- The resulting distribution tells us if our baseline forecasts (or budget) are unrealistic, lets us measure exposure or how much margin is at risk – individually by contract or net position
- Scenarios or alternate calibrations can let us assess a wider range of alternatives

**TESTING**
- We can explore alternate strategies (e.g., “what if” for hedging, contract mix, contract structure) by adding them to the model and exploring the result
- This can be to set targets that optimize a chosen metric
Our value proposition includes custom tools and technology, and rests on our distinctive capabilities in modeling risk and technology.

**MODEL WITH ORDINARY SPREADSHEETS**

Our simulation engine lets us build models in Excel or Google Sheets and translate them to Monte Carlo simulations. Get the power and flexibility of custom-coded models, without sacrificing the ease, transparency, and re-use of spreadsheets.

**EXPLORE & VISUALIZE ANYWHERE**

Models can be run from data analysis tools like Excel, R or Python for ease of exploration. They can be compiled to Javascript and run in the browser or in the cloud¹, to embed into webpages or interactive dashboards.

**HANDLE UNLIMITED COMPLEXITY**

We can handle the most complex contracts, risk factors and complex analyses. Models can be translated to fast C code and run in the cloud¹, and are easy to embed as part of complex multi-stage analyses (e.g., stochastic optimization).

¹ We securely deploy to the cloud – our internal tools run on the Amazon clouds, with popular containerization technology (Docker) and appropriate security (e.g., encryption at rest, VPN)
MODEL WITH SPREADSHEETS

1. Start with a driver-based spreadsheet model – similar to building an income statement as a function of the key risk drivers

2. Plug-in driver calibrations to model (with our own @Risk-like extensions)

3. Spreadsheet is translated to our internal language for Monte Carlo simulation (easy to extend)

4. Internal language is compiled to fast code that can be run on its own

5. Run the code from R/Python or preliminary dashboards to analyze, visualize, and form insights

6. Re-use (JS) code to create interactive dashboards for exploring insights

7. Re-use same code to construct more complex analyses
CALIBRATION

We calibrate simulation risk drivers with a mix of statistics and common sense, to ensure dependable simulation outputs.

**Statistical Based on Historical Data**
- Statistical calibration of several stochastic processes (GBM, MOU) based on historical data – take into account stability of calibration with changes in input time range
- Distribution should fit well, and also reflect intuitive or theoretical attributes of process (e.g., mean-reverting or not, positive or not, floors/caps)

**Forward-Looking Based on Futures**
- Market’s view on volatility (/distribution) implied by market prices for options on futures
- Can sanity-check volatilities of statistically calibrated distribution vs. implied volatilities, or re-calibrate by “moment fitting” (e.g., adjust coefficient to get the desired volatility)

**Business Sanity-Check**
- Results must be believable to the business in order to be used – e.g., broadly align with business’ assumptions of future trends
- Can sanity-check results of calibration, or use assumptions to ad-hoc adjust selected parameters (e.g., set future means for mean-reversion to a baseline forecast)

Using multiple approaches to calibrate and sanity-check drivers can provide confidence that the results are plausible – this confidence is critical for the final results to be trusted.
TECHNOLOGIES WE USE

**PROGRAMMING LANGUAGES**
We pick the right language for the job, from data munging to real-time OLAP
- **Python**
- **JavaScript (incl. ES6, TypeScript)**
- **C/C++ (incl. "modern" C++11/14)**
- **Rust**
- **Julia**
- **OCaml**

**DATA SCIENCE TOOLS**
We develop our own data-analysis tools while taking advantage of existing ones

**WEB & DATA SERVERS**
We use the right tools for storing and analyzing data, and for

**INFRASTRUCTURE & DEVOPS**
We run our long-running calculations and back-end tasks in the cloud on AWS, using Docker to coordinate deployment between dev and production

**SECURITY**
We follow security best-practices\(^1\) for confidential data

---
\(^1\) Including 2FA, encryption-at-rest for potentially confidential data, isolating calculations on sensitive data to private networks, and secure credentials management for sensitive access tokens and passwords.