Ki Insurance - Property Binder Portfolio Optimisation

Abstract

The Lloyd's of London specialty insurance market is ripe for disruption. Global risk landscapes are changing at an unprecedented pace, and insurers must surely embrace algorithmic approaches to remain competitive.

We present a novel application of modern portfolio theory within the context of specialty insurance. We first construct a deterministic loss simulation for our existing property binder portfolio, utilising the output of external catastrophe and in-house attritional loss models. The high-dimensionality of the portfolio and complex non-linear relationships captured within the simulation introduce a constraint-optimisation problem which is hard to solve using traditional methods. Hence, we reach for the class of biology-inspired stochastic optimisation techniques and implement genetic algorithms [1, 2] to optimise our portfolio for metrics including expected return and volatility. To the best of our knowledge, there has been no previous application of evolutionary algorithms for portfolio management within the insurance industry.

Keywords:

Constraint Optimisation Problem, Modern Portfolio Theory, Monte Carlo Simulation, Genetic Algorithms.

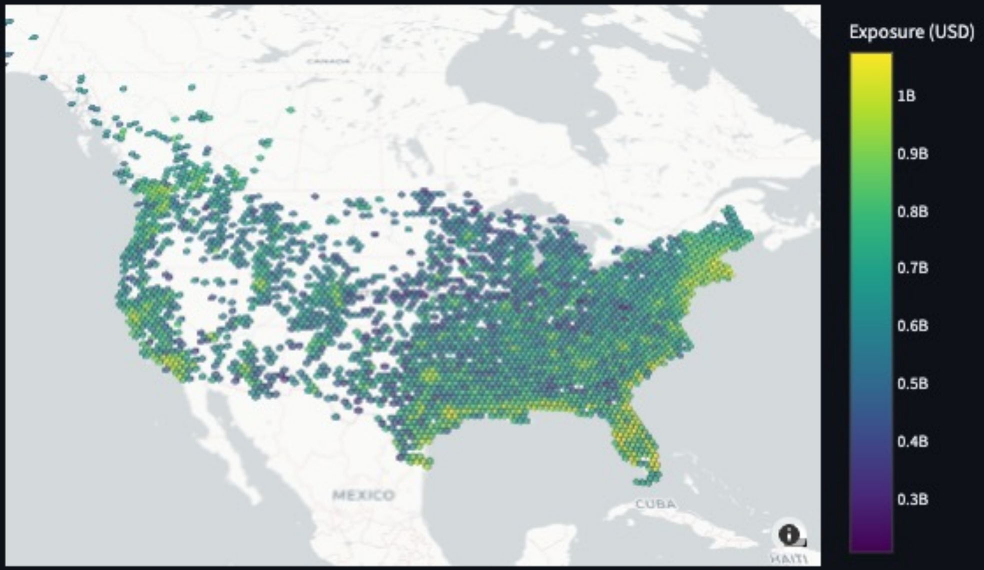
References:

1. Holland, J.H. (1975). Adaptation in natural and artifcial systems. The U. of Michigan Press.
2. Alam, T., Qamar, S., Dixit, A., and Benaida M. (2020). Genetic Algorithm: Reviews, Implementations, and Applications.

Context

- Ki Insurance is the first fully digital and algorithmically driven syndicate operating within the Lloyd"s of London market.
- A proportion of our book consists of United States property binders, primarily covered for hurricane and earthquake perils, and Ki is uniquely posititioned in the market to deploy flexible line sizes.
- We model this portfolio using catastrophe and attritional loss models over 10,000 years of events, capturing correlations between policies where they overlap geographically.

Property Binders Portfolio Exposure



Disclaimer: all data in this presentation has been randomised to maintain confidentiality.

Simulation

1. Let us construct a deterministic loss model for the property binders portfolio as follows:

Superfluous Mathematical Description...

Let $\mathcal{B}$ be the set of all binder policies in the portfolio simulated over the set of all years $\mathcal{Y}$.

For policy $b \in \mathcal{B}$, let l_b and p_b be the line size allocation and total premium respectively.

For policy $b \in \mathcal{B}$ in year $y \in \mathcal{Y}$, let $c_{b,y}$ be the sum of the total catastrophe and attitional claims.

Denote $r_{b,y}$ to be the return for policy $b \in \mathcal{B}$ in year $y \in Y$, hence,

$$r_{b,y} = l_b(p_b - c_{b,y}).$$

Therefore, the portfolio return in year $y \in Y$ is,

$$r_y = \Sigma_{b \in \mathcal{B}} r_{b,y}.$$

The portfolio mean return μ is given by,

$$\mu = \frac{\Sigma_{y \in \mathcal{Y}} r_y}{|\mathcal{Y}|}.$$

The portfolio volatility σ is given by,

$$\sigma = \sqrt{\frac{\Sigma_{y \in \mathcal{Y}}(r_y - \mu)^2}{|\mathcal{Y}|}}.$$

The portfolio Sharpe ratio ϕ is given by,

$$\phi = \frac{\mu}{\sigma}.$$

The portfolio TVaR metric, for given percentile p, τ_p is given by,

$$\tau_p = \frac{\Sigma_y(r_y)}{|\mathcal{Y}|(1-p)},$$

for years $y \in \mathcal{Y}$ such that the returns r_y are in the lowest $(1-p)$ percentile.

The portfolio total premium ρ is given by,

$$\rho = \Sigma_{b \in \mathcal{B}} p_b.$$

The portfolio mean loss ratio θ is given by,

$$\theta = 1 - \frac{\mu}{\rho}.$$

2. We proceed to introduce constraints on the portfolio space, such that each binder policy can be signed between a minimum and maximum line size allocation determined by an underwriter.

3. Let us now perform a Monte Carlo simulation, generating portfolios by uniformly sampling line sizes between the policy constraints and evaluating each portfolio for metrics using the deterministic loss model.

Simulation

Input **Number of Policies:**

1

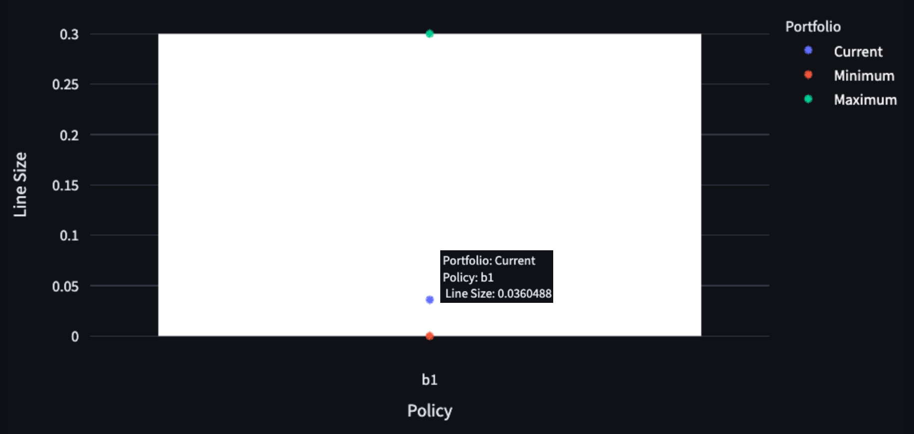
1 100

Input **Number of Random Portfolios:**

0

0 200

Monte Carlo Portfolio Line Sizes



Monte Carlo Simulation Results

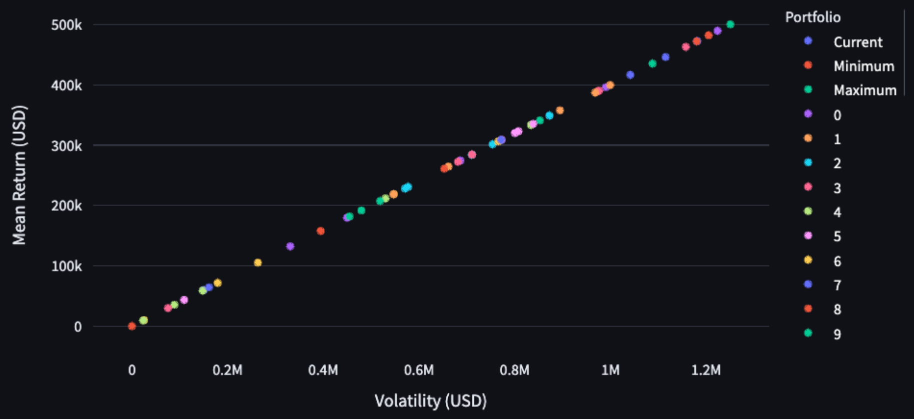
Simulation

Input Number of Policies: 1 (slider 1–100)

Input Number of Random Portfolios: 50 (slider 0–200)

Monte Carlo Portfolio Line Sizes

Monte Carlo Simulation Results



Simulation

Input **Number of Policies:**

2

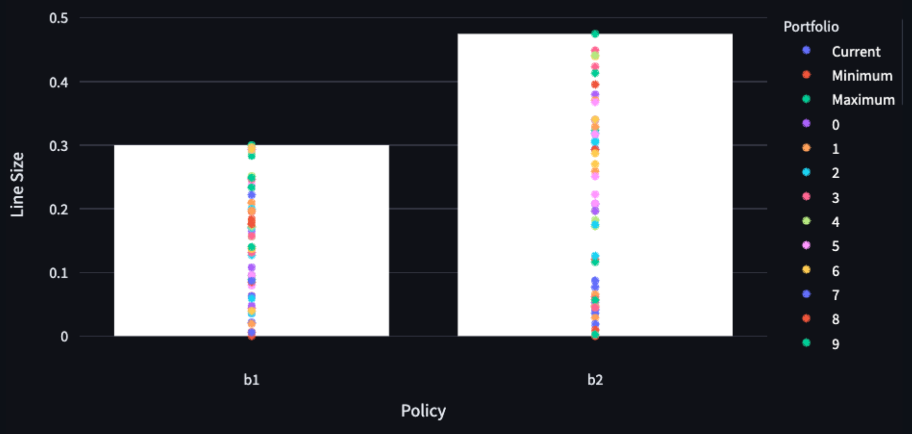
1 100

Input **Number of Random Portfolios:**

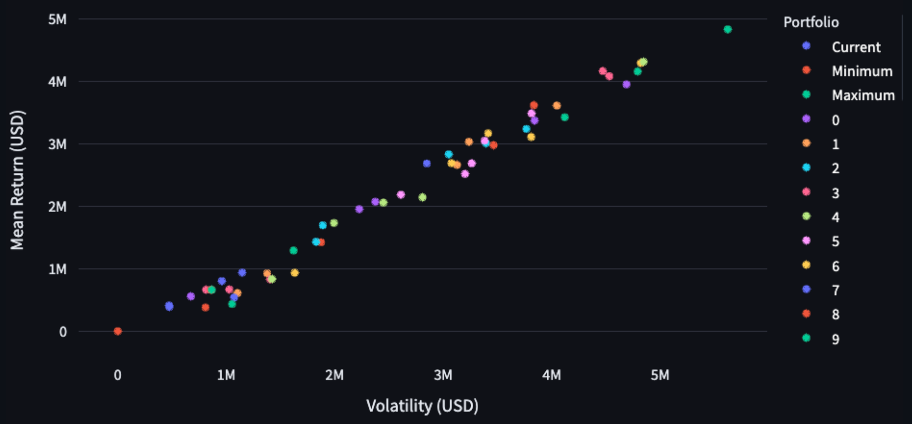
50

0 200

Monte Carlo Portfolio Line Sizes



Monte Carlo Simulation Results



Simulation

Input Number of Policies:

20

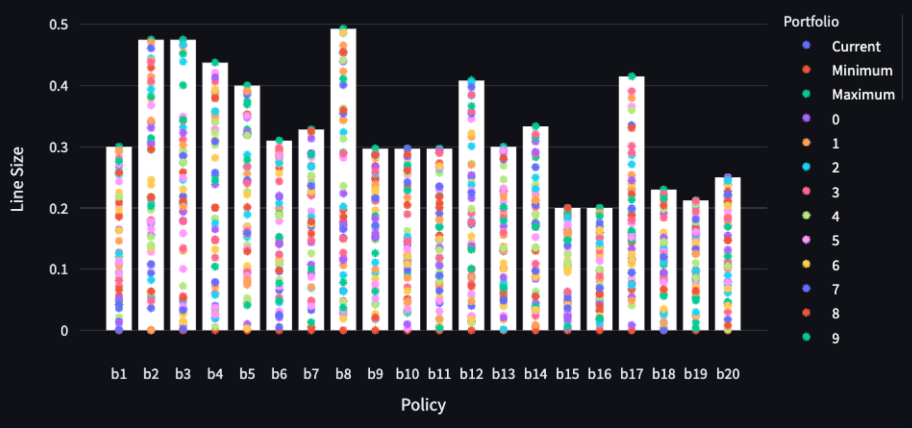
1 100

Input Number of Random Portfolios:

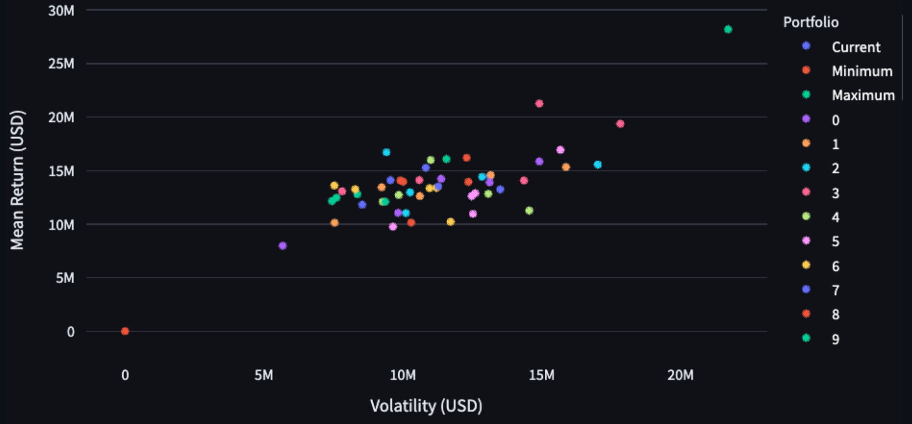
50

0 200

Monte Carlo Portfolio Line Sizes



Monte Carlo Simulation Results



Portfolio Optimisation

Fitness Function

Let us phrase our portfolio simulation and constraints in the context of an optimisation problem.

Consider six objective functions $\mu, \sigma, \phi, \tau, \rho$, and θ optimising mean return, volatility, sharpe ratio, TVaR95, premium, and mean loss ratio respectively.

Define our fitness function $F(\underline{w})$, for some weight vector $\underline{w}$, to be some linear combination of the objective functions which we seek to maximize. Hence,

$$F(\underline{w}) = w_1\mu + w_2\sigma + w_3\phi + w_4\tau + w_5\rho + w_6\theta,$$

Portfolio Optimisation

Genetic Algorithms

Given the high-dimensionality of the constraints and the non-linear nature of the simulation, we decide to use stochastic optimisation methods to optimise our portfolio.

Furthermore, we opt for a population based approach as seek to find a diverse set of "optimal" portfolios to provide our underwriters with multiple strategies.

Thus, we implement genetic algorithms as per the pseudocode below:

```python
def genetic_algorithm(verbose=True):
    # Initialise population
    population = initialise_population()

    # Iterate over generations
    for generation in range(n_generations):
        # Simulate population
        metrics = simulate(population)

        # Evaluate fitness distribution
        distribution = fitness_function(metrics)

        # Select fittest individuals
        elite_population = selection(population, distribution)

        # Create offspring
        succession_population = crossover(population, distribution)
        succession_population = mutation(succession_population)

        # Construct next generation population
        population = elite_population + succession_population

    if verbose:
        print("fin.")
    return population
```

Portfolio Optimisation

Demonstration

We propose the following scenario and algorithm parametrisation:

Example Setup.

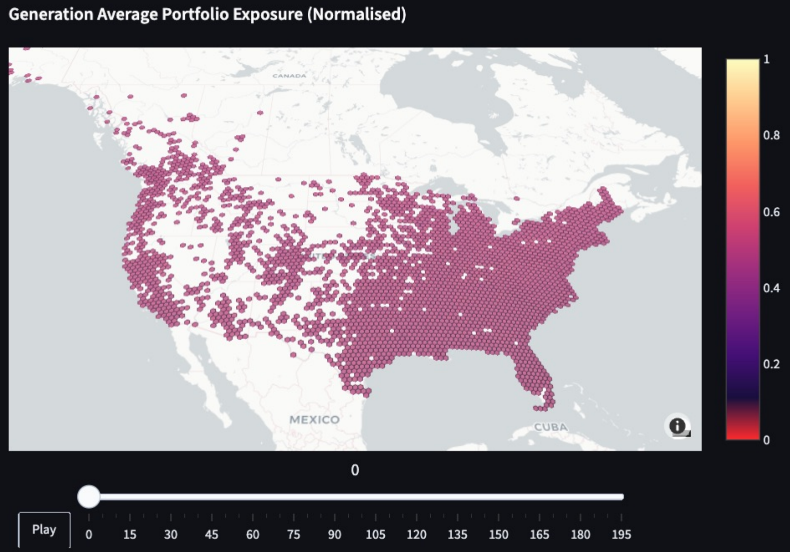
- The portfolio simulation consists of 100 binder policies with arbitrary line size constraints.
- The population is initialised with 1,000 random portfolios and iterated over 200 generations.
- The fitness function is defined to maximise Sharpe ratio and minimise distance to a target $200m premium.
- The genetic algorithm is parametrised such that each new generation of portfolios is constructed using a 20% elitism rate and an 80% succession rate.
- The succession offspring are generated by drawing pairs of parent portfolios from the softmax population fitness distribution, from which the line sizes are swapped following a 5% crossover rate and randomised at a 2% mutation rate.

Portfolio Optimisation

Training Visualisation

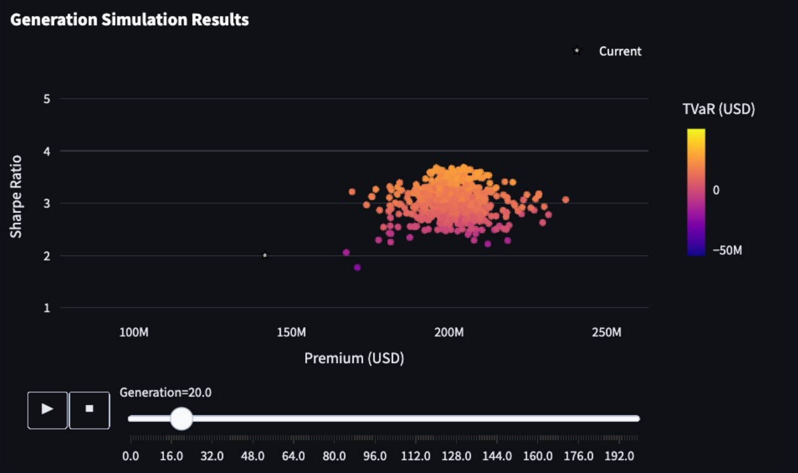
Using Uber's H3 geospatial indexing system, we aggregate portfolio exposure into hexagons and are able to plot the population average exposure optimising across the United States.

Note, we normalise the exposure of each hexagon independently for visual interpretability.



Portfolio Optimisation

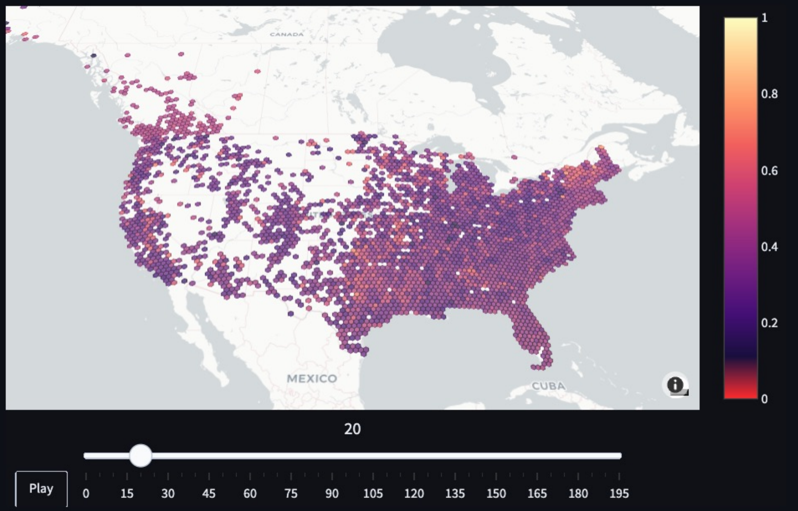
Training Visualisation



Using Uber's H3 geospatial indexing system, we aggregate portfolio exposure into hexagons and are able to plot the population average exposure optimising across the United States.

Note, we normalise the exposure of each hexagon independently for visual interpretability.

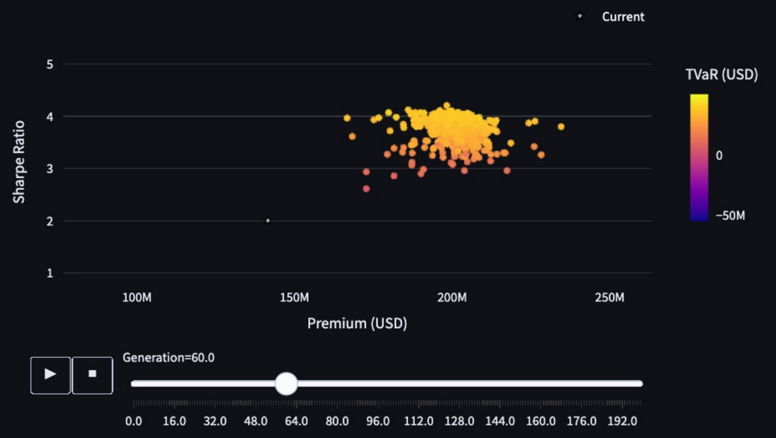
Generation Average Portfolio Exposure (Normalised)



Portfolio Optimisation

Training Visualisation

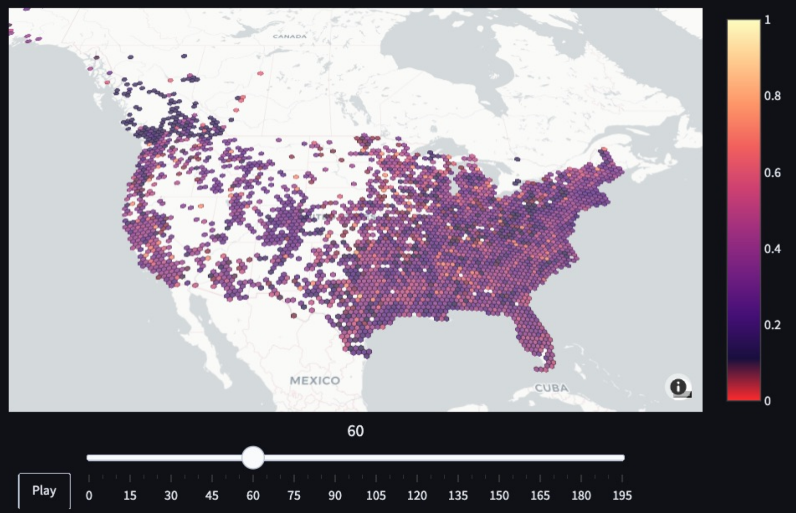
Generation Simulation Results



Using Uber's H3 geospatial indexing system, we aggregate portfolio exposure into hexagons and are able to plot the population average exposure optimising across the United States.

Note, we normalise the exposure of each hexagon independently for visual interpretability.

Generation Average Portfolio Exposure (Normalised)



Portfolio Optimisation

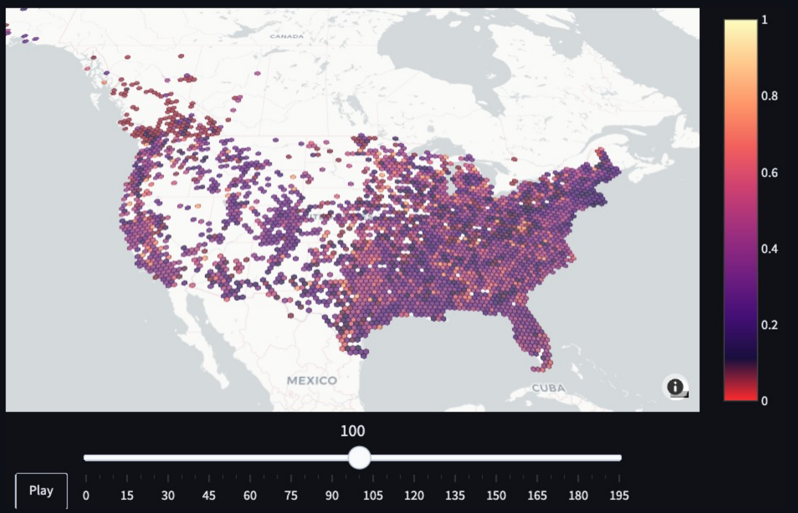
Training Visualisation

Generation Simulation Results

Using Uber's H3 geospatial indexing system, we aggregate portfolio exposure into hexagons and are able to plot the population average exposure optimising across the United States.

Note, we normalise the exposure of each hexagon independently for visual interpretability.

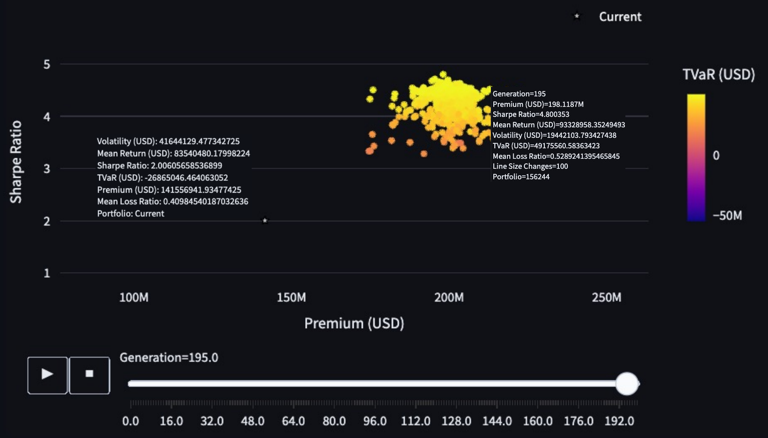
Generation Average Portfolio Exposure (Normalised)



Portfolio Optimisation

Training Visualisation

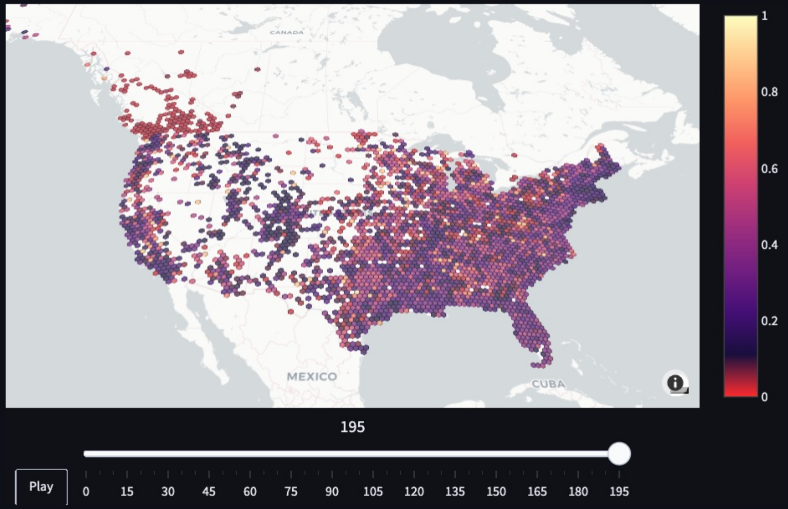
Generation Simulation Results



Using Uber's H3 geospatial indexing system, we aggregate portfolio exposure into hexagons and are able to plot the population average exposure optimising across the United States.

Note, we normalise the exposure of each hexagon independently for visual interpretability.

Generation Average Portfolio Exposure (Normalised)



Portfolio Optimisation

Optimal Results

The genetic algorithm outputs a large number of potential solutions that may be challenging to interpret.

Hence, we begin to reduce the pool of portfolios by selecting those along the efficient frontier, offering the best mean return to volatility tradeoff.

We continue to filter the portfolios to determine our "optimal" portfolios with a premium within $5m from the target premium and a volatility less than $25m.

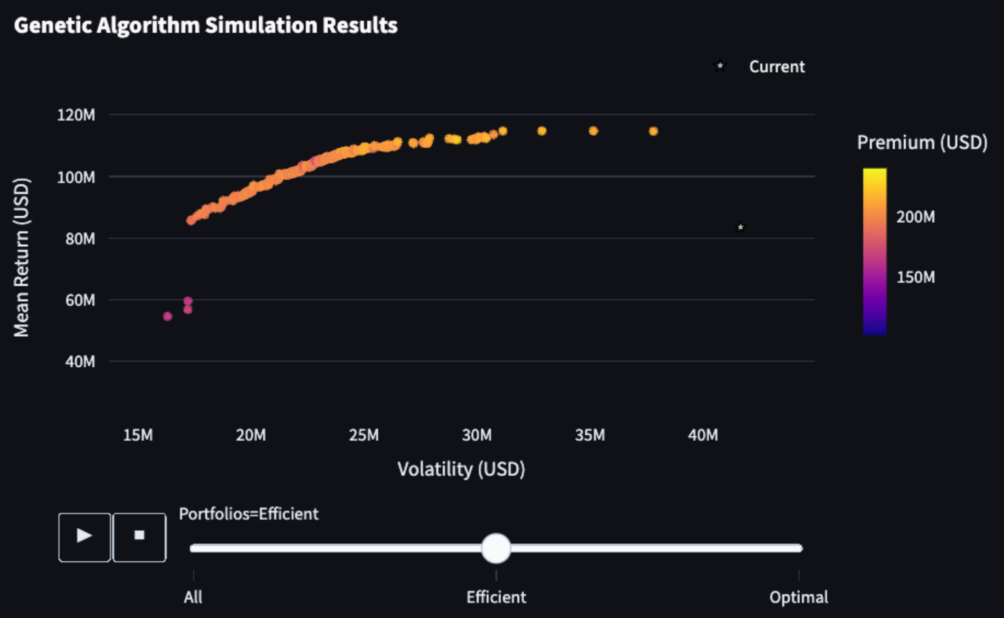
Portfolio Optimisation

Optimal Results

The genetic algorithm outputs a large number of potential solutions that may be challenging to interpret.

Hence, we begin to reduce the pool of portfolios by selecting those along the efficient frontier, offering the best mean return to volatility tradeoff.

We continue to filter the portfolios to determine our "optimal" portfolios with a premium within $5m from the target premium and a volatility less than $25m.



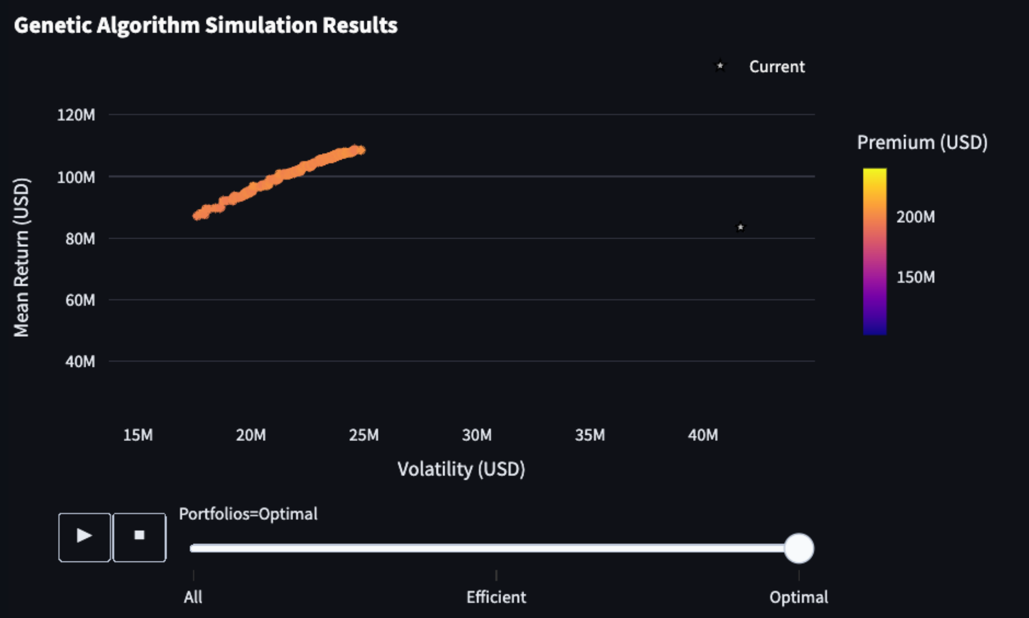
Portfolio Optimisation

Optimal Results

The genetic algorithm outputs a large number of potential solutions that may be challenging to interpret.

Hence, we begin to reduce the pool of portfolios by selecting those along the efficient frontier, offering the best mean return to volatility tradeoff.

We continue to filter the portfolios to determine our "optimal" portfolios with a premium within $5m from the target premium and a volatility less than $25m.



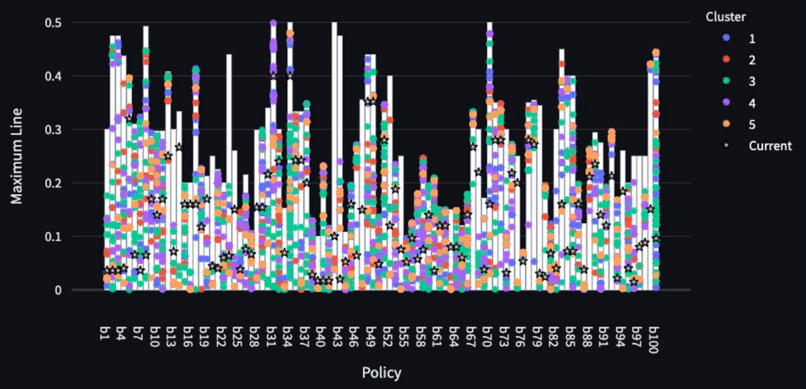
Portfolio Optimisation

Clustering Analysis

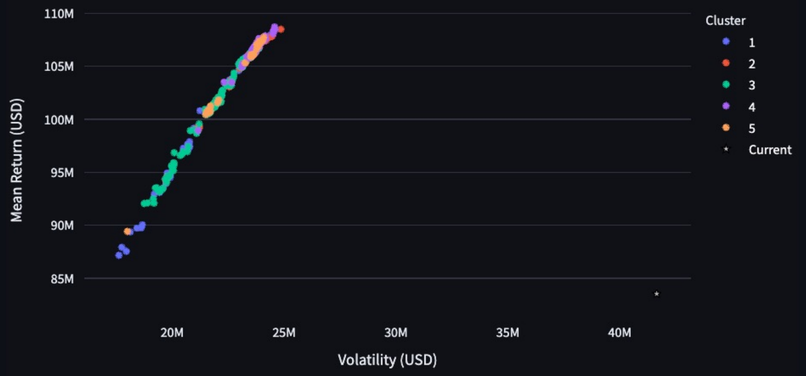
In order to further simplify these "optimal" portfolios, we now perform hierarchical clustering in the portfolio space to yield distinct groups of portfolio compositions with desireable performance metrics.

These clusters allow underwriters flexibility when choosing a strategy, before they then consider factors that are not captured as part of the simulation- for example, applying a commercial lens for line size viability, hotspot monitoring, and historic performance.

Clustered Optimal Portfolio Line Sizes



Clustered Optimal Portfolio Simulation Results



Portfolio Optimisation

Clustering Analysis

In order to further simplify these "optimal" portfolios, we now perform hierarchical clustering in the portfolio space to yield distinct groups of portfolio compositions with desireable performance metrics.

These clusters allow underwriters flexibility when choosing a strategy, before they then consider factors that are not captured as part of the simulation- for example, applying a commercial lens for line size viability, hotspot monitoring, and historic performance.

Clustered Optimal Portfolio Line Sizes

Clustered Optimal Portfolio Simulation Results

Insurance Data Science Conference 2024.

Adam Bedwell-Smith, adam.bedwell-smith@ki-insurance.com.

Finlay Duff, finlay.duff@ki-insurance.com.