Single Zone Evaluation

José I. Velarde Morales

August 19, 2022

1 Setup

Data Generating Process

We use the following data generating processes for the toy examples. In order to make the comparison with the status quo more straightforward, we model losses instead of overall wealth.

$$
\begin{aligned}
l &= \beta\theta + \epsilon \\
\theta &\sim \mathcal{N}(5, 2) \\
\epsilon &\sim \mathcal{N}(0, 1) \\
\beta &= 3
\end{aligned}
$$

I draw 100 training samples from the above model to train the prediction model and to use as input for the optimization programs. I then evaluate both methods using 1000 samples drawn from the same model.

2 Status Quo

We will be comparing our proposed approach to the method developed by Chantarat et al. (2013), which, to the extent of our knowledge is what is currently being used for Kenya's Index Based Livestock Insurance (IBLI) program. The method is as follows:

- First, they use a clustering algorithm to group locations into clusters
- They then use historical data to fit a linear regression model to predict herd mortality rates in each cluster. They fit a different model for each cluster.
- Contracts are of the form: $I(\theta) = \max(\hat{M}(\theta) - M^*, 0) \times TLU \times P_{TLU}$ where $\hat{M}(\theta)$ is the predicted herd mortality rate, M^* is the strike value, TLU is the number of insured livestock units, and P_{TLU} is the price per insured livestock unit. In other words, their contract pays farmers for the full predicted loss beyond a threshold, M^*. This threshold, M^* is the contract's strike value.
- They choose the strike value that would explain the highest share of insurable losses in the historical data. Specifically, they run the following regression: $y_s = \beta_s \hat{y_s} + \epsilon$ where y_s is the actual insured losses at strike value s and $\hat{y_s}$ is the predicted insured losses at strike value s. For example, suppose that $TLU = 100$ (ie there are 100 insured units), and that $P_{TLU} = 25$ (ie each unit is worth 25), and that $M^* = 0.25$ (ie contract starts paying out once the predicted mortality rate exceeds 25%). If the actual mortality rate is 0.5, then actual

insured losses would be $y_{25} = \max(M - M^*, 0) \times TLU \times P_{TLU} = (0.5 - 0.25) \times (100) \times (25)$. If the predicted mortality rate in that scenario was 0.4, the predicted insured losses, $\hat{y_{25}} = \max(\hat{M}(\theta) - M^*, 0) \times TLU \times P_{TLU} = (0.4 - 0.25) \times (100) \times (25)$. They use historical data to calculate $y_s, \hat{y_s}$, and then run the following regression: $y_s = \beta_s \hat{y_s} + \epsilon$. They choose the strike value $s = \arg\max_s \beta_s$. This takes into account the fact that the prediction model, $\hat{M}(\theta)$ might be better at predicting some losses better than others.

To mimick this in our toy example, we set the status quo contracts to be $I(\theta) = \max(\hat{l}(\theta) - l^*, 0)$, since we are already assuming that l is the total loss suffered. For the toy example, we fit a (correctly specified) linear regression model to predict losses: $l = \beta\theta + \epsilon \implies \hat{l}(\theta) = \hat{\beta}\theta$.

3 Optimization Approach

In order to separate the effect of contract design from the effect of prediction quality, we will be basing our contracts on the same predictions used by the status quo method. In other words, we will use the status quo method to estimate a model that predicts loss based on theta, $\hat{l}(\theta)$, and our payout function will use that as input instead of θ. In other words, our model will define payout functions $I(\hat{l}(\theta))$, where $\hat{l}(\theta)$ is the same prediction function used by the status quo method.

Minimum CVaR Model

This model minimizes the $CVaR$ of the farmer's net loss subject to a constraint on the premium. The premium constraints are expressed as a fraction of the full insured amount.

Model Parameters

- ϵ: This defines the CVaR objective. $\epsilon = 0.1$ means that our objective is on the expected value of the loss given that it is above the 90^{th} percentile.
- $\bar{\pi}$: This is the maximum value of the premium.
- y: maximum insured amount

Single Zone Model

$$\min_{a,b \geq 0} \; CV@R_{1-\epsilon}\left(\ell - \min\left\{(a\theta + b), K\right\}\right) \tag{1}$$

$$\text{s.t.} \quad a\mathbb{E}\left[\theta\right] + b \leq \bar{\pi} \tag{2}$$

$$a\theta + b \geq 0 \tag{3}$$

We reformulated the problem in the following way. In the model below, p_k is the probability of event k, and k indexes the possible realizations of θ, l.

$$\min_{a,b,\pi,\gamma,t} \quad t + \frac{1}{\epsilon} \sum_k p_k \gamma_k \tag{4}$$

$$\text{s.t. } \gamma_k \geq l^k - \min\left\{(a\hat{l}(\theta^k) + b), K\right\} - t, \forall k \tag{5}$$

$$\gamma_k \geq 0, \forall k \tag{6}$$

$$0 \leq a\hat{l}(\theta^k) + b, \forall k \tag{7}$$

$$K\bar{\pi} \geq a\mathbb{E}[\hat{l}(\theta)] + b \tag{8}$$

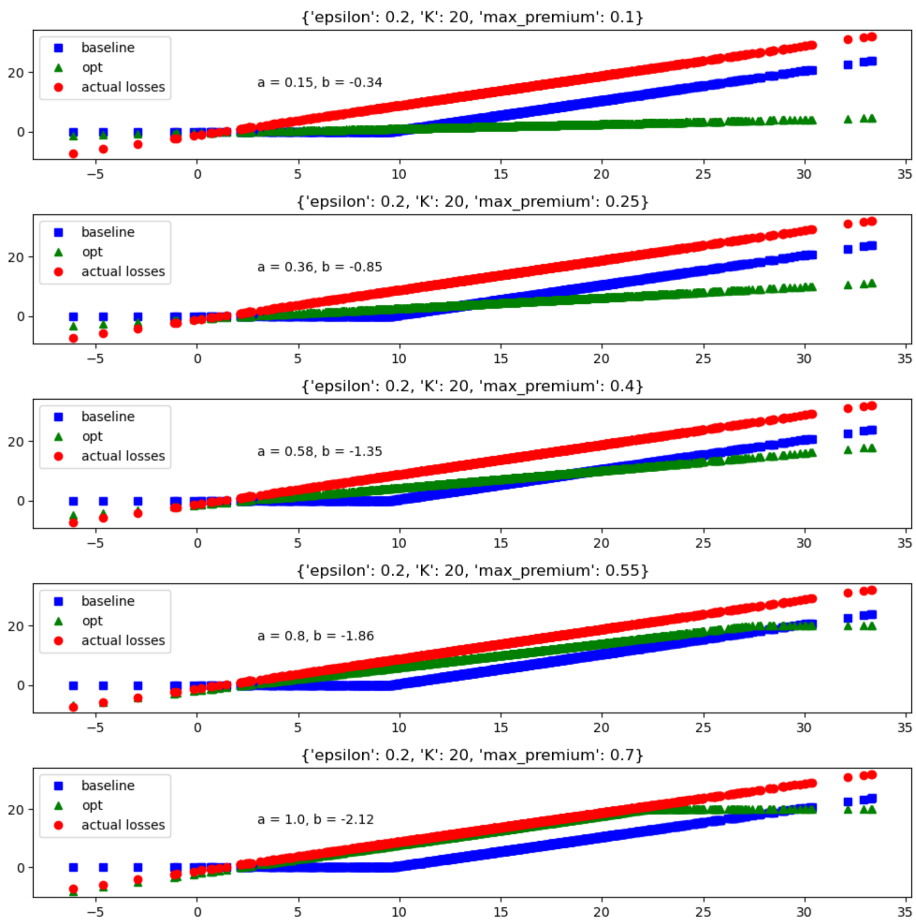
4 Results

The model mostly behaves as expected. Increasing the payment cap, K increases a and decreases b, in other words, it increases the slope of the payout function and shifts it right. Similarly, increasing the maximum premium, $\bar{\pi}$ increases the payout rate and shifts it right as well. However, a and b didn't seem sensitive to ϵ at all. I think this is because the data generating process is linear, $\ell = \beta\theta + \epsilon$. When I replace that data generating process with a cubic one, $\ell = \beta\theta^3 + \epsilon$, a, b become sensitive to ϵ. In general, as ϵ decreases, a increases and b decreases, which I think is the behavior we would expect, because it would be more targeted towards losses at the right tail of the distribution.

CVaR Model

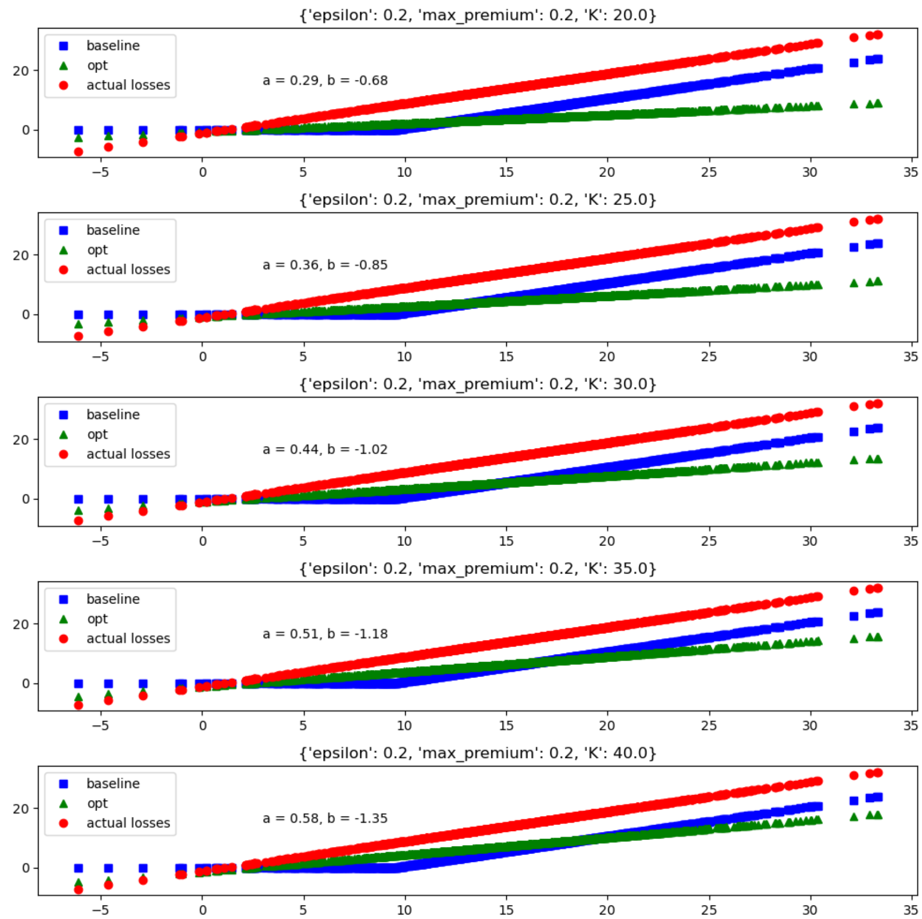
Max Premium Exploration

Figure 1: Max Premium Exploration



K Exploration

Figure 2: K Exploration



Epsilon Exploration

Figure 3: Epsilon Exploration