On the Comparative Performance of Machine Learning and Economic Models for Risky Decision-Making

Mateus Hiro Nagata

HEC Paris

November 20, 2024

Motivation

- Model selection: simplicity and descriptiveness;
- Understanding risk attitudes required specific choice of utility $u(\cdot)$ and weighting probability function $w(\cdot)$;
- By pre-specifying the functional form, we risk overlooking meaningfully observable consistencies, or we might end up capturing irrelevant parameters:

Simplicity

- Occam's razor;
- Model complexity and overfitting;
- Induction problem (Harman & Kulkarni, 2012).

Contribution

- Study decision under risk by **predicting** certainty equivalents using economic, machine learning, and hybrid models;
- Hybrid models: neural networks with economic insights;
- Interpretable ML vs Black-box models;
- **Technical contribution**: a method for eliciting both utility and probability distortions *simultaneously*.

Results

- Economic model (CPT-MLE) is the best among the explainable;
- No additional gain with more flexible $u(\cdot), w(\cdot)$;
- Prediction at the expense of interpretation;
- (Unsurprisingly) individual identification is useful;
- "Opening the black-box" leads to economic models.

Advantages of the Method

- Does not need the data to be adaptive and made for this;
- Does not have a layered formula that may induce errors;
- Does not assume a parametric form;
- Simultaneously elicit utility and weights;
- Easily interpretable but maybe not explainable.

Related Literature

- Prediction of decision under risk and ML: (Athey & Imbens, 2019; Ellis et al., 2023; Erev et al., 2010, 2017; Fudenberg et al., 2022; Peterson et al., 2021; Peysakhovich & Naecker, 2017);
- Elicitation: (Abdellaoui, 2000; Perny et al., 2016; Vieider et al., 2015);
- Risk/Preference Heterogeneity: (Bruhin et al., 2010; Vieider et al., 2015).

Closest ones

- "Regression problem" and "open" the black-box (Peysakhovich & Naecker, 2017) ;
- Methodology closer to Peterson et al. (2021) but does not solve the problem of estimating inverse utility (regression vs classification problem).

Binary Prospects - Choice

Binary prospect

$$(x, p; y).$$

- $x \geq y \geq 0$, for gains, $x \leq y \leq 0$ for losses;
- $x \geq 0 \geq y$ for mixed prospects;
- Outcome x with probability p and y with probability $1 - p$;
- x, y, ce in euros;

What is the sure amount ce such that

$$(x, p; y) \sim ce$$

$$f(\mathbf{x}) = ce.$$

$$(x, p; y) \sim ce$$

$$w(p)u(x) + (1 - w(p))u(y) = u(ce)$$

$$u^{-1}(w(p)u(x) + (1 - w(p))u(y)) = ce$$

$$f(\mathbf{x}) = ce$$

Data

Table 1: Positive Prospects

Prospect	Equivalent
$(5, \frac{1}{2}; 0)$	ce_1
$(10, \frac{1}{2}; 0)$	ce_2
$(20, \frac{1}{2}; 0)$	ce_3
$(30, \frac{1}{2}; 0)$	ce_4
$(30, \frac{1}{2}; 10)$	ce_5
$(30, \frac{1}{2}; 20)$	ce_6
$(20, \frac{1}{8}; 0)$	ce_7
$(20, \frac{1}{8}, 5)$	ce_8
$(20, \frac{2}{8}, 0)$	ce_9
$(20, \frac{3}{8}, 0)$	ce_{10}
$(20, \frac{5}{8}, 0)$	ce_{11}
$(20, \frac{6}{8}, 0)$	ce_{12}
$(20, \frac{7}{8}, 0)$	ce_{13}
$(20, \frac{7}{8}, 5)$	ce_{14}

Table 2: Negative Prospects

Prospect	Equivalent
$(-5, \frac{1}{2}; 0)$	ce_{15}
$(-10, \frac{1}{2}; 0)$	ce_{16}
$(-20, \frac{1}{2}; 0)$	ce_{17}
$(-20, \frac{1}{2}; -5)$	ce_{18}
$(-20, \frac{1}{2}; -10)$	ce_{19}
$(-20, \frac{1}{8}; 0)$	ce_{20}
$(-20, \frac{1}{8}; -5)$	ce_{21}
$(-20, \frac{2}{8}; 0)$	ce_{22}
$(-20, \frac{3}{8}; 0)$	ce_{23}
$(-20, \frac{5}{8}; 0)$	ce_{24}
$(-20, \frac{6}{8}; 0)$	ce_{25}
$(-20, \frac{7}{8}; 0)$	ce_{26}
$(-20, \frac{7}{8}; -5)$	ce_{27}

Table 3: Mixed Prospect

Prospect	Equivalent
$(20, \frac{1}{2}; y_{28})$	0

Quality of Data

- Heterogeneous population (30 countries, 2939 subjects);
- Incentivized choice;
- Data from Vieider et al. (2015);
- Holt and Laury's (2002) choice list method.

Metrics

- Prediction rule $f: X \to Y$;
- Let P be the joint distribution over inputs $\mathbf{x} \in X$ and certainty equivalents $ce \in Y$, and let P_X, P_Y be their respective marginals;
- Loss function $\ell : Y \times Y \to \mathbb{R}$, $\ell(y, y') = (y' - y)^2$;

Mean Squared Error (MSE)

$$\mathcal{E}_P(f) = \mathbb{E}_P[\ell(f(\mathbf{x}), ce)].$$

Model Comparison - Completeness

How well does the model (family) $\mathcal{F}_\Theta$ predict?

- Eligible set $\mathcal{F} : X \to Y$;
- f_{base} = Naive baseline;
- f^* = Best-fit model in the feasible set of functions $\mathcal{F}$;
- f^*_Θ = Best-fit model in the parametric family. $\mathcal{F}_\Theta := \{f_\theta\}_{\theta \in \Theta}$;
- Completeness of a model (family) $\mathcal{F}_\Theta$ is

$$\frac{\mathcal{E}_P(f_{base}) - \mathcal{E}_P(f^*_\Theta)}{\mathcal{E}_P(f_{base}) - \mathcal{E}_P(f^*)}.$$

Model Comparison - Restrictiveness

How well does our model family capture a noisy f?

- Discrepancy function

$$d(f, f') = \mathbb{E}_{P_X}[\ell(f(\mathbf{x}), f'(\mathbf{x}))];$$

- Let $\lambda_{\mathcal{F}}$ be the uniform distribution over the feasible set of functions $\mathcal{F}$;
- $d(\mathcal{F}_\Theta, f) := \arg\min_{\theta \in \Theta} d(f_\theta, f)$;
- Restrictiveness of a model $\mathcal{F}_\Theta$ is

$$\frac{\mathbb{E}_{\lambda_{\mathcal{F}}}[d(\mathcal{F}_\Theta, f)]}{\mathbb{E}_{\lambda_{\mathcal{F}}}[d(f_{base}, f)]}.$$

Model Selection

Ideally, we want a model that is:

- Interpretable, with completeness $= 1$, and restrictiveness $= 1$;
- If not possible, at least we can define a "Pareto-frontier".

Cumulative Prospect Theory - MLE

$$CPU = \begin{cases} w^{+}(p)u(x) + (1 - w^{+}(p))u(y) \text{ if } x, y \geq 0 \\ w^{-}(p)u(x) + (1 - w^{-}(p))u(y) \text{ if } x, y \leq 0 \\ w^{+}(p)u(x) + w^{-}(1 - p)u(y) \text{ if } x > 0, y < 0 \end{cases}$$

$$CPU = \pi_x u(x) + \pi_y u(y),$$

Cumulative Prospect Theory - MLE

$$u(x) = \begin{cases} x, \text{ if } x > 0; \\ -\lambda(-x) \text{ if } x \leq 0. \end{cases}$$

$$w^{+}(p) = \exp(-\beta^{+}(-\ln(p))^{\alpha^{+}})$$

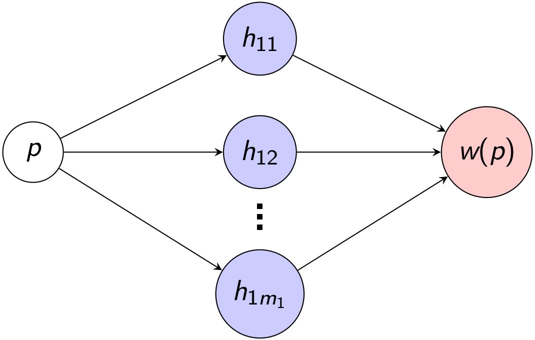
$$w^{-}(p) = \exp(-\beta^{-}(-\ln(p))^{\alpha^{-}}),$$

Neural Networks

Neural Network as a universal function approximator (Hornik et al., 1989). For example,

1. Input: p;
2. Neural Network's output: $w_{params}(p)$;
3. Prediction: $\hat{ce} = w_{params}(p)x + (1 - w_{params}(p))y$;
4. Loss function: $(\hat{ce} - ce)^2$;
5. Alter NN's parameters to decrease loss function.

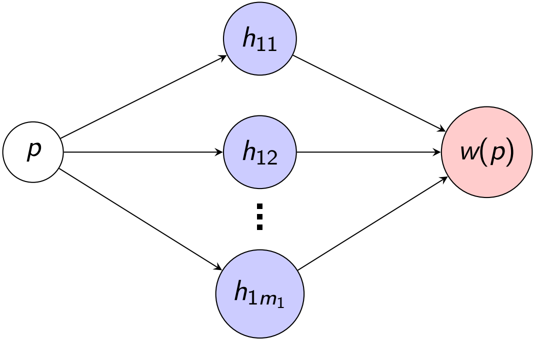
Neural Network for Neural RDU (Probability Weighting)



$$h_{1k} = \sigma(w_1 p + b_{1k}), k = 1, \ldots, m_1$$

$$\sigma(x) = \max\{0, x\}$$

Neural Network for Neural RDU (Probability Weighting)

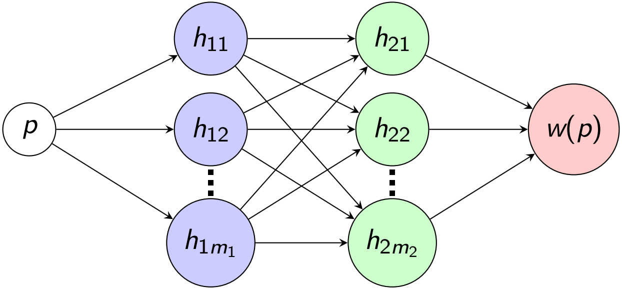


$$\mathbf{h}_1 = \sigma(\mathbf{w}^1 p + \mathbf{b}^1)$$

$$w(p) = \sigma(\mathbf{w}^2 \mathbf{h}_1 + b^2)$$

$$\hat{ce} = w(p)x + (1 - w(p))y$$

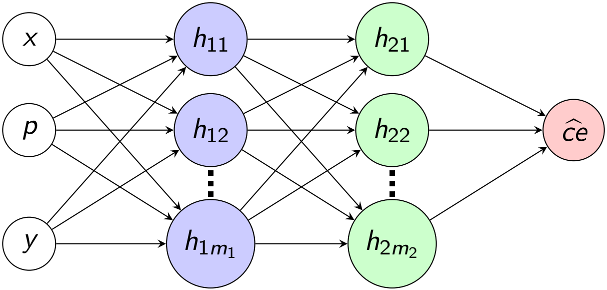
$$\underset{\mathbf{w}^1,\mathbf{w}^2,\mathbf{b}^1,b^2}{\arg\min}\ \mathbb{E}_P[(\hat{ce} - ce)^2].$$



$$\mathbf{h}_1 = \sigma(\mathbf{w}^1 p + \mathbf{b}^1)$$

$$\mathbf{h}_2 = \sigma(\mathbf{W}^2 \mathbf{h}_2 + \mathbf{b}^2)$$

$$w(p) = \sigma(\mathbf{w}^3 \mathbf{h}_2 + b^3)$$



$$\mathbf{x} = [x\ p\ y]^T$$

$$\mathbf{h}_1 = \sigma(\mathbf{W}^1\mathbf{x} + \mathbf{b}^1)$$

$$\mathbf{h}_2 = \sigma(\mathbf{W}^2\mathbf{h}_2 + \mathbf{b}^2)$$

$$\hat{ce} = \sigma(\mathbf{w}^3\mathbf{h}_2 + b^3) = NN(\mathbf{x})$$

Neural Networks

1. Neural RDU (Linear): $\hat{ce} = w(p)x + (1 - w(p))y$;
2. Neural CPT (Linear): $\hat{ce} = \pi^x x + \pi^y y$;
3. Neural RDU: $\hat{ce} = u^{-1}[w(p)u(x) + (1 - w(p))u(y)]$;
4. Neural CPT: $\hat{ce} = u^{-1}[\pi^x u(x) + \pi^y u(y)]$;

5. NN: $\hat{ce} = NN(x, y, p)$;
6. NN with ID: $\hat{ce} = NN(x, y, p, id)$;
7. Neural RDU with ID: $\hat{ce} = NN(w(p)x + (1 - w(p))y, id)$.

The Problem of Inversion

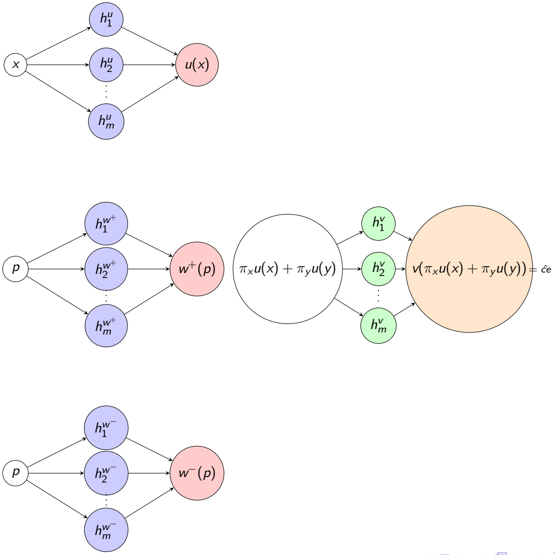
Problem: difficult to estimate a function and its inverse.
Solution:

- Neural network for $u(\cdot)$, $w^{+}(\cdot)$, $w^{-}(\cdot)$, $v(\cdot)$, where $v = u^{-1}$;
- Goal: estimate $\hat{ce} = v[\pi^x u(x) + \pi^y u(y)]$;
- Constrained optimization intuition:

$$\begin{aligned} \underset{\mathbf{w}^1,\mathbf{W}^2,\mathbf{w}^3,\mathbf{b}^1,\mathbf{b}^2,b^3}{\text{minimize}} \quad & \mathbb{E}_P[(\hat{ce} - ce)^2] \\ \text{subject to} \quad & v(u(x)) = x, \ \forall x \in X. \end{aligned}$$

Loss Function Trick

$$\underset{\mathbf{w}^1,\mathbf{W}^2,\mathbf{w}^3,\mathbf{b}^1,\mathbf{b}^2,b^3}{\text{minimize}} \quad \mathbb{E}_P[(\hat{ce} - ce)^2] + \alpha \mathbb{E}_P[(v(u(x)) - x)^2].$$


x
h_1^u
h_2^u
h_m^u
$u(x)$
p
$h_1^{w^+}$
$h_2^{w^+}$
$h_m^{w^+}$
$w^+(p)$
$\pi_x u(x) + \pi_y u(y)$
h_1^v
h_2^v
h_m^v
$v(\pi_x u(x) + \pi_y u(y)) = \hat{ce}$
p
$h_1^{w^-}$
$h_2^{w^-}$
$h_m^{w^-}$
$w^-(p)$

Tree-Based Algorithms

Tree-Based Algorithms

- Random Forests: $rf(x, y, p)$
- XGBoost: $XGBoost(x, y, p)$
- CatBoost: $CatBoost(x, y, p)$

Interpretable Models

1. EVT: $\hat{ce} = px + (1-p)y$;
2. CPT MLE: $\hat{ce} = \pi^x u(x) + \pi^y u(y)$;
3. Neural RDU (Linear): $\hat{ce} = w(p)x + (1 - w(p))y$;
4. Neural CPT (Linear): $\hat{ce} = \pi^x x + \pi^y y$;
5. Neural RDU: $\hat{ce} = v[\pi^x u(x) + \pi^y u(y)]$;
6. Neural CPT: $\hat{ce} = v[\pi^x(p)u(x) + \pi^y u(y)]$.

Black-Box Models

id is embedded in a high-dimensional space

7. NN: $\hat{ce} = NN(x, y, p)$;
8. NN with ID: $\hat{ce} = NN(x, y, p, id)$;
9. Neural RDU with ID (Linear): $\hat{ce} = NN(w(p)x + (1 - w(p))y, id)$;
10. Random Forest: $\hat{ce} = RF(x, y, p)$;
11. XGBoost: $\hat{ce} = XGBoost(x, y, p)$;
12. CatBoost: $\hat{ce} = CatBoost(x, y, p)$.

Model	**MSE**	**MAE**	**Completeness**	**Restrictiveness**
EVT	19.85	3.04	0	1
CPT MLE	15.87	2.92	0.78	0.59
Neural RDU (Linear)	16.63	2.97	0.64	0.52
Neural CPT (Linear)	16.37	2.98	0.69	0.46
Neural RDU	16.62	3.03	0.64	0.45
Neural CPT	15.62	2.92	0.84	0.39
$NN(x, p; y)$	15.81	2.92	0.80	0.36
Random Forest	14.81	2.74	1	0.25
XGBoost	14.81	2.74	1	0.25
CatBoost	14.81	2.74	1	0.25

Table: Predictive performance of aggregate economic and machine learning models.

Individual Identification

Model	**MSE**	**MAE**	**Completeness**	**Restrictiveness**
$NN(x, p; y, id)$	10.28	2.22	1	0.53
$NN(w(p)x + (1 - w(p))y, id)$	14.93	2.85	0.48	0.47

Table: Predictive performance of individual economic and machine learning models.

Interesting Results

General:

- No gain in functional freedom for NN;
- Loss-aversion, differences in weighting in gains and losses exist;
- All tree-based models have similar performance and fare the best, but may be too flexible;
- Presence of noise that cannot be captured by only $(x, p; y)$ as inputs.

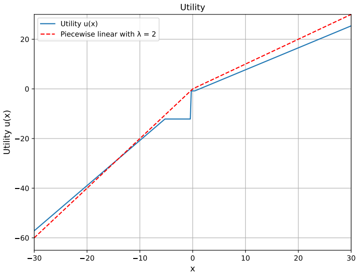


Figure: Utility - Neural CPT.

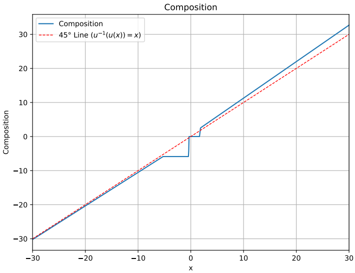


Figure: Composition - Neural CPT.

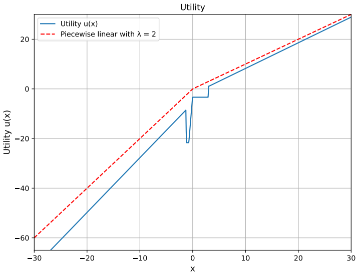


Figure: Utility - Neural RDU

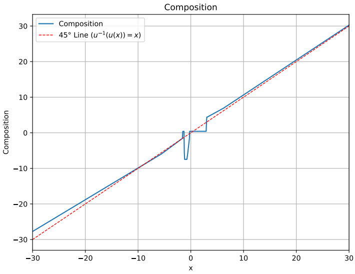


Figure: Composition - Neural RDU

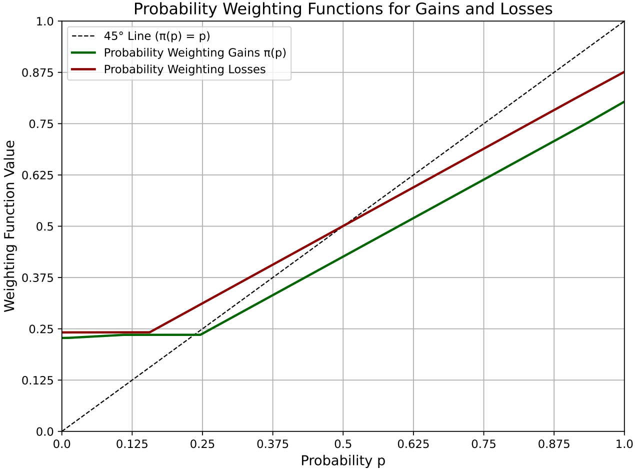


Figure: Weighting - Neural CPT.

Remark

- Not explicitly requiring $w(\cdot)$ to be a probability function, or $u(\cdot)$ to be monotonic, or the functions to be simple;
- Simplest function that includes certainty effect and possibility effect.

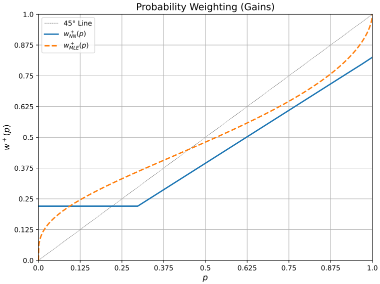


Figure: Probability Weighting for Gains of CPT Models.

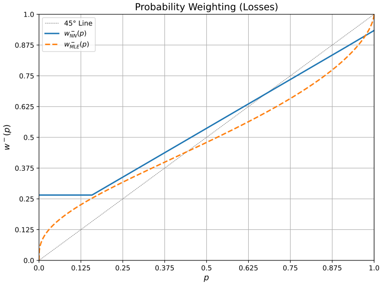
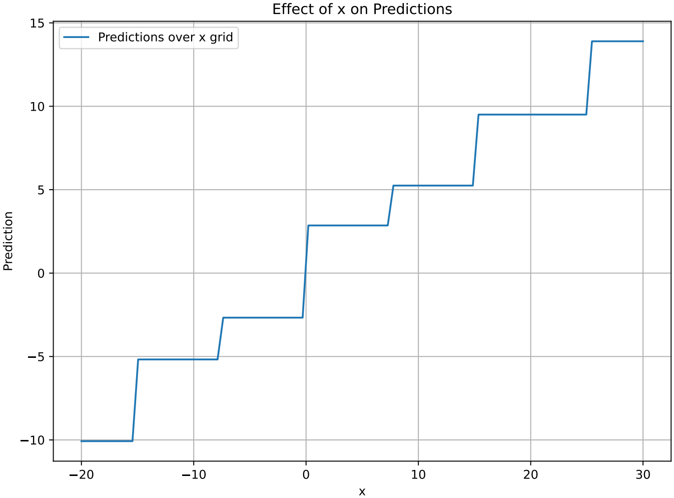


Figure: Probability Weighting for Losses of CPT Models.

Effect of Marginal Change in Forests



Data

Table 1: Positive Prospects

Prospect	Equivalent
$(5, \frac{1}{2}; 0)$	ce_1
$(10, \frac{1}{2}; 0)$	ce_2
$(20, \frac{1}{2}; 0)$	ce_3
$(30, \frac{1}{2}; 0)$	ce_4
$(30, \frac{1}{2}; 10)$	ce_5
$(30, \frac{1}{2}; 20)$	ce_6
$(20, \frac{1}{8}; 0)$	ce_7
$(20, \frac{1}{8}, 5)$	ce_8
$(20, \frac{2}{8}, 0)$	ce_9
$(20, \frac{3}{8}, 0)$	ce_{10}
$(20, \frac{5}{8}, 0)$	ce_{11}
$(20, \frac{6}{8}, 0)$	ce_{12}
$(20, \frac{7}{8}, 0)$	ce_{13}
$(20, \frac{7}{8}, 5)$	ce_{14}

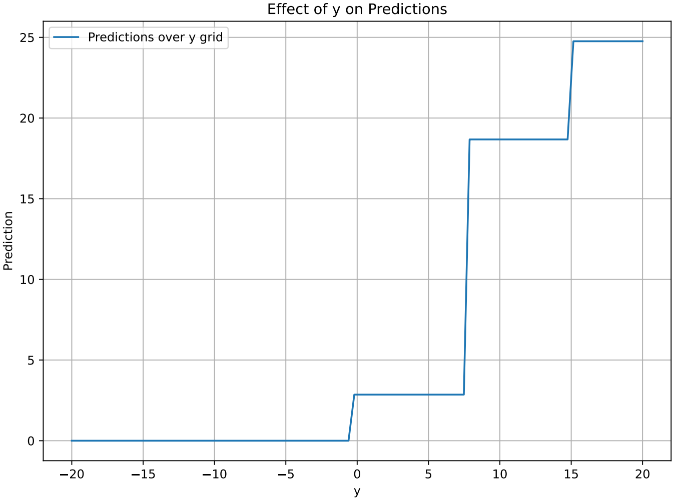
Table 2: Negative Prospects

Prospect	Equivalent
$(-5, \frac{1}{2}; 0)$	ce_{15}
$(-10, \frac{1}{2}; 0)$	ce_{16}
$(-20, \frac{1}{2}; 0)$	ce_{17}
$(-20, \frac{1}{2}; -5)$	ce_{18}
$(-20, \frac{1}{2}; -10)$	ce_{19}
$(-20, \frac{1}{8}; 0)$	ce_{20}
$(-20, \frac{1}{8}; -5)$	ce_{21}
$(-20, \frac{2}{8}; 0)$	ce_{22}
$(-20, \frac{3}{8}; 0)$	ce_{23}
$(-20, \frac{5}{8}; 0)$	ce_{24}
$(-20, \frac{6}{8}; 0)$	ce_{25}
$(-20, \frac{7}{8}; 0)$	ce_{26}
$(-20, \frac{7}{8}; -5)$	ce_{27}

Table 3: Mixed Prospect

Prospect	Equivalent
$(20, \frac{1}{2}; y_{28})$	0

Effect of Marginal Change in Forests



Conclusion

- Simplest models that incorporate possibility and certainty effect, loss-aversion, and domain-dependent weighting probability are optimal;
- There is a notion of proximity of individuals' risk attitudes;
- Hybrid methods are useful to check the presence of any non-explained variance;
- Prediction can be improved by sacrificing interpretation and at the risk of overfitting.

References I

- Abdellaoui, M. (2000). Parameter-free elicitation of utility and probability weighting functions. *Management science*, *46*(11), 1497–1512.
- Athey, S., & Imbens, G. W. (2019). Machine learning methods that economists should know about. *Annual Review of Economics*, *11*(1), 685–725.
- Bruhin, A., Fehr-Duda, H., & Epper, T. (2010). Risk and rationality: Uncovering heterogeneity in probability distortion. *Econometrica*, *78*(4), 1375–1412.
- Ellis, K., Kariv, S., & Ozbay, E. (2023). The predictivity of theories of choice under uncertainty.

References II

Erev, I., Ert, E., Plonsky, O., Cohen, D., & Cohen, O. (2017).From anomalies to forecasts: Toward a descriptive model of decisions under risk, under ambiguity, and from experience.. *Psychological review*, *124*(4), 369.

Erev, I., Ert, E., Roth, A. E., Haruvy, E., Herzog, S. M., Hau, R., Hertwig, R., Stewart, T., West, R., & Lebiere, C. (2010).A choice prediction competition: Choices from experience and from description. *Journal of Behavioral Decision Making*, *23*(1), 15–47.

Fudenberg, D., Kleinberg, J., Liang, A., & Mullainathan, S. (2022).Measuring the completeness of economic models. *Journal of Political Economy*, *130*(4), 956–990.

References III

Harman, G., & Kulkarni, S. (2012). *Reliable reasoning: Induction and statistical learning theory*. MIT press.

Holt, C. A., & Laury, S. K. (2002).Risk aversion and incentive effects. *American economic review*, *92*(5), 1644–1655.

Hornik, K., Stinchcombe, M., & White, H. (1989).Multilayer feedforward networks are universal approximators. *Neural networks*, *2*(5), 359–366.

Perny, P., Viappiani, P., & Boukhatem, A. (2016).Incremental preference elicitation for decision making under risk with the rank-dependent utility model. *Uncertainty in Artificial Intelligence*.

References IV

Peterson, J. C., Bourgin, D. D., Agrawal, M., Reichman, D., & Griffiths, T. L. (2021). Using large-scale experiments and machine learning to discover theories of human decision-making. *Science*, *372*(6547), 1209–1214.

Peysakhovich, A., & Naecker, J. (2017). Using methods from machine learning to evaluate behavioral models of choice under risk and ambiguity. *Journal of Economic Behavior & Organization*, *133*, 373–384.

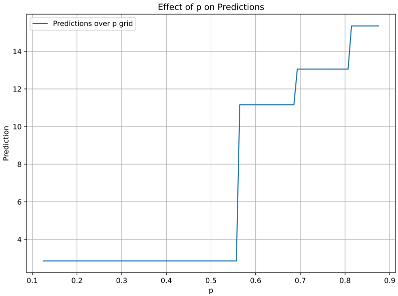
References V

Vieider, F. M., Lefebvre, M., Bouchouicha, R., Chmura, T., Hakimov, R., Krawczyk, M., & Martinsson, P. (2015). Common components of risk and uncertainty attitudes across contexts and domains: Evidence from 30 countries. *Journal of the European Economic Association*, *13*(3), 421–452.

Parameter	α^+	β^+	α^-	β^-	λ	σ
Estimate	0.60 (0.01)	0.90 (0.02)	0.64 (0.04)	0.94 (0.09)	1.96 (0.02)	0.22 (0.01)

Table: Parameter estimates of CPT MLE

Weightings Forests



Weightings

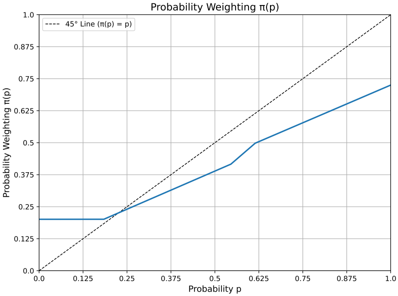


Figure: Weighting - Neural RDU (Linear)

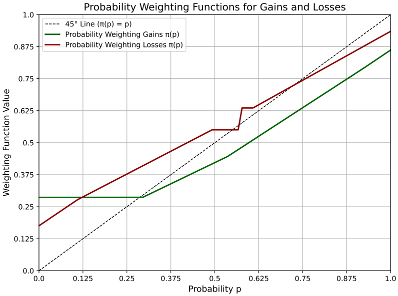


Figure: Weighting - Neural CPT (Linear)

Weighting Neural RDU

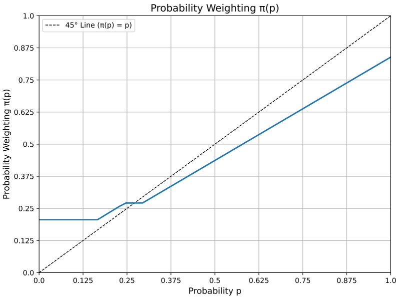


Figure: Weighting - Neural RDU

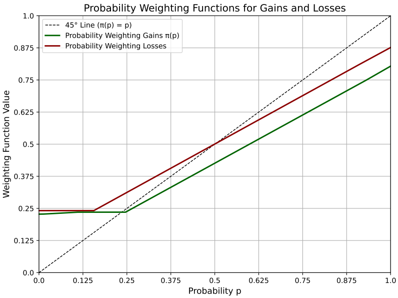
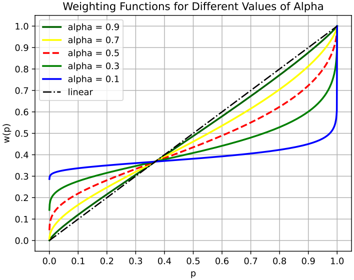
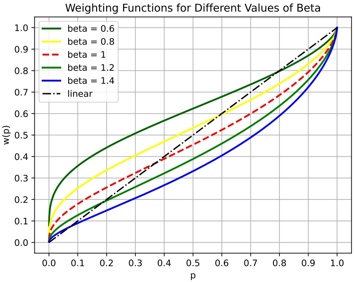


Figure: Weighting - Neural CPT

Prelec's Compound Invariance Family



Prelec's Compound Invariance Family



MLE

$$CPU_i = w^+(p_i)x_i + (1 - w^+(p_i))y_i, \;\; x_i, y_i \geq 0$$
$$CPU_i = w^-(p_i)x_i + (1 - w^-(p_i))y_i, \;\; x_i, y_i \leq 0$$

$$\hat{y}_i = \frac{w^+(p_i)x_i}{\lambda w^-(1 - p_i)}, x_i \geq 0 \geq y_i$$

To deal with the noise in the data, assume $ce_i = CPU_i + \varepsilon_i, \varepsilon_i \sim \mathcal{N}(0, \sigma_i^2)$.

$$\psi(\theta, (x_i, p_i; y_i)) = \phi\left(\frac{CPU_i(\alpha^+, \alpha^-, \beta^+, \beta^-, \lambda, \sigma) - ce_i}{\sigma}\right)$$

$$LL(\theta, \gamma) = \sum_{i=1}^{N} log[\psi(\theta, (x_i, p_i; y_i))]$$

Tree-Based Algorithms

Intuitively,

1. Tree making: Split feature space X (for instance, for $x \geq 15$, predict average $E[y|x \geq 15]$ and $E[y|x \leq 15]$ optimally and recursively, defining a tree $T_1(\mathbf{x})$
2. Define trees $T_1, \ldots, T_B$ by selecting random subsets of data and features
3. Ensemble those trees. Predict the average $\sum_{b=1}^{B} T_b(x) = T(x)$

Random Forest Algorithm

Go Back

Algorithm Random Forest Algorithm

for $b = 1, \ldots, B$ **do**

a) Draw a bootstrap sample $\mathbf{Z}^*$ of size N from the training data

b) Grow a random-forest tree T_b to the bootstrapped data by recursively repeating the following steps for each terminal node of the tree until the minimum node size n_{min} is reached:

- Select m variables at random from the p variables
- Pick the best variable/split-point among the m
- Split the node into two daughter nodes

Output: $\hat{f}_{\text{rf}}^{B}(x) = \frac{1}{B}\sum_{b=1}^{B} T_b(x)$

XGBoost Algorithm (Part 1)

Algorithm XGBoost Algorithm

Input: Training data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, regularization parameters λ, γ

Parameters: Number of trees T, learning rate η, max depth d, minimum samples per leaf n_{min}

for $t = 1, \ldots, T$ **do**

a) Compute gradients and hessians for each data point

b) Build a tree T_t using gradient-based splitting:

- i. For each node, choose the best split point to maximize the gain using the gradient and hessian information
- ii. Apply regularization to avoid overfitting

XGBoost Algorithm (Part 2)

Algorithm XGBoost Algorithm (continued)

for $t = 1, \ldots, T$ **do**

1:

c) Update the predictions $\hat{y}_t$:

$$\hat{y}_t = \hat{y}_{t-1} + \eta T_t(x)$$

2:

3: **Output:** The ensemble of trees $\{T_t\}_{t=1}^{T}$

4: **Prediction at a new point** x**:**

$$\hat{f}(x) = \sum_{t=1}^{T} \eta T_t(x)$$

CatBoost Algorithm (Part 1)

Algorithm CatBoost Algorithm

Input: Training data $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, categorical feature support

for $t = 1, \ldots, T$ **do**

a) Apply ordered boosting to prevent target leakage

b) Handle categorical features using target statistics and a prior

c) Build a tree T_t:

- i. Use gradient-based loss and regularization
- ii. Apply symmetry constraints to handle categorical features

$=0$

On the Comparative Performance of Machine Learning and Economic Models for Risky Decision-Making

Mateus Hiro Nagata

HEC Paris

November 20, 2024