Multi-Stage Solicitation of Probability Distributions

Experiments, Theory and Perspective on Conformal Prediction

Peter Cotton

Working draft v0.3 · 2026

Abstract

The microprediction platform built distributional prediction supply chains, chaining pools so that one's output became the message or the settled outcome of the next. This note is a self-critical, ex-post look at the game theory of those chains: the games were built first, and here we ask, of each, whether truthful reporting was really the best move. A chain is proper where an exogenous outcome anchors every stage; a downstream stage can then be scored on its own residual or, for the log score, at the top level. By that test the games mostly hold up, one of them only by a lucky accident. Flaws aside, the setup has continuing pedagogical merit, as it illustrates the inherent limitation of conformal prediction.

1 The games that were built

Adam Smith's pin factory (Smith 1776) made pins cheaply by splitting the work into specialized steps. Prediction can be organized the same way. A forecast that has to be produced over and over is cheaper if it is broken into standard sub-forecasts, each contributed by whoever is best at it, the supply chain the microprediction project set out to build (Cotton 2019, 2022). Markets and contests reward competition, and competition makes any single sub-forecast cheap, but it does not assemble the sub-forecasts into anything: a market pays for the best forecast of its own number and then stops, with no reason to pass its output along.

Chaining can help. If the output of one mechanism becomes the message, or the settled outcome, of the next, the separate contests link into a pipeline and the supply chain takes shape. However, we know of exactly one platform that chained probabilistic outputs this way: the retired microprediction platform, which ran several such chains at once. The only near-relative, Manifold's resolves-to-market markets, settles on another market's price with no exogenous outcome anywhere in the chain, the invalid unanchored case (§2). Chaining on point outputs is common, but that is derivative structure, not composition of probabilistic elicitation.

Conformal prediction is a closer cousin. It has drawn wide attention lately, and it is exactly a prescription for residual modeling, which we read here as a restricted, invitation-only variety of chained market (§4).

The microprediction platform. A *stream* was a live quantity that someone published to the platform one value at a time: airport wait times, the electricity load for New York State, emoji usage on Twitter. The base game was to forecast the next value of a stream before it arrived. A forecaster submitted 225 Monte Carlo scenarios, a fixed-size sample standing for her predictive distribution. When the value landed, the pot was split in proportion to how near her scenarios fell to it: the nearest-the-pin rule. A contributor who bets to maximize her long-run wealth does best under it to place her scenarios according to her honest distribution, the log-optimal all-in Kelly bet (Cotton 2022); a contributor optimizing something else stakes differently, so truthfulness here is the log-optimal player's property, not the pot split's.

The base game settled only a single scalar, but its output could be transformed into fresh scalars, and those became games of their own. Take the residual first. Once the community has forecast a stream, re-express each outcome on a community-relative scale: its percentile, or the z-score got from that percentile by the inverse normal transform.[1][2] The z-score opened its own stream, `z1~`: forecast the distribution of the next one. A participant who can say more about it than the community can, because the community is biased, is the wrong shape, or is blind to a covariate she holds, is paid for the difference. In the language of §2 this is a residual stage, eliciting the law of that z-score.

Dependence used the same idea. The joint behaviour of two streams is two-dimensional, but a pool settles one number. Each stream's z-score was mapped back to a uniform on $[0, 1]$ by the normal CDF, the two uniforms were interleaved into one by the Morton z-curve (interleave the binary digits of the two coordinates, the way a geohash packs latitude and longitude into one string), and the result was put back through the inverse normal transform to give `z2~`, a pool on which is a market on the copula of the pair. Three streams folded the same way gave `z3~`. A 2020 copula contest ran exactly this on the five-minute comovements of five cryptocurrencies, contributors submitting 225 samples packed through the z-curve (Cotton 2020a).

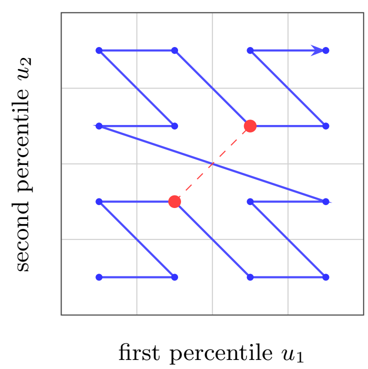

Figure 1: The Morton z-curve threads the unit square of two percentiles into a single line, so a bivariate copula is priced as one univariate pool. The thread is not distance-preserving: the two red points are neighbours in the square but far apart along the curve, so a nearness-based pool on the folded scalar prices a metric the curve chose, not the joint's (§3).[4]

A related stacked-lottery design let competing algorithms contribute monotone maps that composed into one forecast (Cotton 2020b, slides 29-31). The platform is retired.

Its successors. montepredicition is the base game with one stage repeated rather than

[1]The name *z-score* is a terminological overloading where the analogy to the statistical one is loose; nothing here is assumed normal. The number is the community's market-implied law F_1, an arbitrary distributional transform rather than a Gaussian, composed with the inverse normal Φ^{-1}. Because F_1 is the market's implied distribution, it is a risk-neutral z-score.

[2]The streams could equally have been left uniform on $[0, 1]$; the normal scale was chosen because it was felt to make sharp, well-calibrated prediction slightly easier.

[4]The author was aware of the distortion; the z-curve was a practical choice that avoided a major refactoring of the platform.

chained. Each weekly submission is about a million joint scenarios of eleven sector-ETF returns; the score is an exponential-kernel density of the submitted scenarios read at the realised vector, blended with the eleven leave-one-dimension-out marginals, and the pot is split by that score (`monteprediction/scoring.py`) (Cotton 2024c, 2024a). Wealth threads across rounds, and the contest has run since January 2024. The MidOne contests at CrunchDAO priced residual densities directly (CrunchDAO 2024; Cotton 2024b), a residual stage without the sample smoothing.

For contrast, the point-output world. Chaining on point outputs is everywhere: every derivative, the electricity virtual bids and transmission rights written on day-ahead prices (Jha and Wolak 2023), a market settling on another market's reported number. That, though, is ordinary derivative structure, not composition of probabilistic elicitation. A different near-miss puts margins and dependence in parallel rather than in series: the racetrack's win versus exotic pools, index versus single-name option books, tranche versus single-name CDS. These books settle independently, their consistency left to arbitrage (Harville 1973; Hausch et al. 1981), so parallel is not chained and the two can disagree.

2 What can be proven

Each of these games composes stages that share one form. A *stage* is a transducer over one message type: it carries a wealth state, consumes reports that are distributional beliefs, and where it settles takes a realized outcome and returns transfers. A scoring rule, a market maker, a pool, and a parimutuel are all stages, and the operators that build and combine them, together with the convex dictionary behind them, are a companion paper (Cotton 2026a). Two facts govern whether a chain of them is valid.

A chain is proper only where an outcome anchors it. Every score is paid against a realized value or its rank; strip the outcome out and every report profile is an equilibrium of the settlement, so nothing is elicited. A chain whose final stage settles on a real outcome, with a proper score at each stage, is well founded, and truthful reporting is a stagewise equilibrium.

Proposition 1 (single-stage guarantees compose). *Suppose each stage, taken with its inputs and settlement transform fixed, makes the truthful report a best response for that stage's own transfer. Then, provided no participant can move a downstream settlement in which they hold a position, truthful reporting at every stage is a stagewise equilibrium: no participant gains by a deviation confined to one stage. Without that no-cross-position condition the chained game may admit derivative-style manipulation of an upstream settlement.*

Proof. A deviation confined to stage k moves the deviator's payoff through stage k's own transfer, where the single-stage hypothesis makes truth a best response, and through the downstream transfers its output feeds. A deviator who holds no downstream position collects none of the latter, so that propagation is payoff-irrelevant to them and only the stage-k transfer remains. ∎

This is not a Nash equilibrium of the full game: a participant who reports upstream and holds a downstream stake holds a derivative on an upstream settlement, with the usual incentive to distort the underlying (Kumar and Seppi 1992; Jarrow 1994; Hanson and Oprea 2009; Ostrovsky 2012). The platform's chains lived inside this caveat.

The residual stage, and two ways to score it. The operator that chains is the residual: a stage emits an aggregate F_1 for an outcome Y, and a second market elicits the law of the residual z-score $Z = \Phi^{-1}(F_1(Y))$, settling at $z = \Phi^{-1}(F_1(y))$. If F_1 were the truth then $F_1(Y)$ is uniform (the probability integral transform) and Z is standard normal, and whatever structure remains is the second stage's edge.

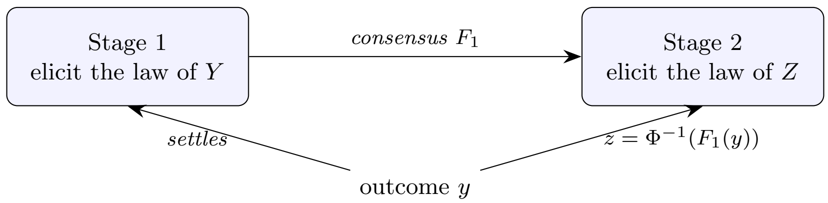

Figure 2: A residual chain. Stage 1's consensus F_1 becomes the frame for stage 2, which settles on the z-score $z = \Phi^{-1}(F_1(y))$; the same realized y anchors both. The base pool and its z_1 residual stream are an instance.

Proposition 2 (the residual correction is a reweighting). *For F_1 strictly increasing with density $p_1 > 0$ and a residual report of density g for the z-score Z, the composed forecast has density $p(y) = p_1(y)\, g(z)/\varphi(z)$ with $z = \Phi^{-1}(F_1(y))$ and φ the standard normal density, so*

$$\log p(y) \;=\; \log p_1(y) + \log \frac{g(z)}{\varphi(z)}.$$

The residual stage is paid $\log g(z) - \log \varphi(z)$, the log-likelihood ratio of its report against the $N(0,1)$ stake, zero when $g = \varphi$.

Proof. On the rank scale the correction is $p(y) = p_1(y)\, h(F_1(y))$ with h the residual density on $[0,1]$ (chain rule); writing the report on the z-scale, $z = \Phi^{-1}(u)$, gives $h(u) = g(z)/\varphi(z)$, hence the displayed forms. The term is the logarithmic score (Cotton 2026a, Thm 1) for the report, strictly proper for the law of Z. ■

The additive form answers a design question raised by the stacked lottery (Cotton 2020b). A contribution to the second market can be judged *locally*, by $\log g(z) - \log \varphi(z)$ on its own residual, or converted to a *top-level* forecast of Y and judged by $\log p(y)$. Proposition 2 makes these the same mechanism, since they differ by $\log p_1(y)$, which no downstream report can move. The equivalence is special to the log score and to scores additive under composition; for a general proper score, local and top-level scoring rank downstream contributions differently.

This is a consideration the long-running debate over log score versus CRPS rarely weighs, since that debate turns on locality, propriety, robustness, and interpretability. For a platform built to chain elicitations, as the microprediction supply chain was (Cotton 2022), the operative question is instead which score lets residual refinements add up. The log score does; CRPS, though proper and a useful CDF-level diagnostic, has no regret that decomposes stage by stage.

Residual chains as boosting. Proposition 2 has each residual stage multiply the running density by a likelihood ratio $g(z)/\varphi(z)$ fitted to what the chain so far gets wrong. That is the greedy stagewise step of boosting under log loss (Mason et al. 1999; Friedman 2001): a chain of residual markets is stagewise gradient boosting, with the market's participants in place of the weak learners and their staked wealth in place of the learning rate. A weak learner need only beat chance; a participant here need only beat the current consensus, and is paid in proportion to how much. The departures from a boosting library are the interesting part. The ensemble is not fit by one optimiser but bid up by a crowd; the shrinkage is endogenous, set by how much wealth backs each correction rather than by a hyperparameter; and there is no explicit regulariser, only the propriety of each stage's score. Whether the iteration converges to the true conditional law, as boosting does under its assumptions, is open (§6).

Dependence factors the same way. By Sklar's theorem (Sklar 1959) a joint density is $\prod_i f_i(x_i) \cdot c(u_1, \dots, u_d)$ in terms of the coordinate marginals $u_i = F_i(x_i)$, so

$$\log p(x) \;=\; \sum_{i=1}^{d} \log f_i(x_i) + \log c(u), \qquad u_i = F_i(x_i).$$

A pipeline of d margin stages and one rank stage settled on the rank vector pays each stage a log score of its own object, and the rank stage's report is strictly proper for the copula given correct margins — the multivariate residual, one dimension per margin.

What the pool actually elicits. The base game does not receive a density; it receives samples and smooths them. That smoothing is where propriety is won or lost.

Proposition 3 (sample elicitation; Cotton (2026c), Thms 1-2). *Score the bandwidth-h kernel density estimate of a submitted cloud by the log score. Settled at the raw outcome, the optimal cloud is drawn from a deconvolution of the belief, not the belief. Settled at the outcome jittered by the same kernel, truthful sampling is optimal, strictly so when the kernel's characteristic function is nonvanishing on a dense set.*

The correction is a jitter of the settlement, and it is the hinge on which the built games turn.

3 Were the games valid?

Take the games of §1 to the tests of §2.

The base pool: approximately proper by lucky accident. The nearest-the-pin pool paid each contributor the density their smoothed samples placed at the realized value. By Proposition 3, scored at the raw outcome that rewards a deconvolution of the belief, not the belief: a contributor whose honest law is Gaussian would be paid most by submitting samples with variance h^2 *below* the truth.

The platform mostly escaped this, but by luck. Discrete outcomes caused computational trouble (ties when the outcome fell on a submitted sample, degenerate estimates), so the implementation jittered the settlement to smooth them over. A jitter of the outcome is the repair Proposition 3 prescribes, and it removed the deconvolution incentive — a fortunate accident, even though the amount added for numerical comfort was not quite equal to the amount matched to the smoothing scale that strict propriety asks. The hack that kept the arithmetic well behaved was the hinge that kept the game roughly honest.

monteprediction is the same rule in eleven continuous dimensions, and here the accident is not available: its score reads an exponential-kernel density of the submitted scenarios at the raw realised vector, with no jitter. By Proposition 3 that is the improper case, unrepaired — the reward tilts, by about the kernel scale, toward scenarios a little tighter than the forecaster's honest ones. The leave-one-dimension-out blend in the score guards against high-dimensional fragility but not this tilt, and a matched jitter, or a fair finite-sample correction, would remove it. This is a live contest, so the point is not academic.

The `z1~` residual stream: valid, and sharp-rewarding. Predicting the realized z-score is a proper elicitation of its law, anchored by the exogenous outcome (with the same jitter caveat as the base pool). It pays for any edge over the community: a forecaster who conditions on a covariate the base game ignored forecasts the z-score's conditional law, beats the community's flat one, and collects the missed information as log-wealth growth, at the rate $I(R;X)$ — the mutual information between the residual R and the conditioning covariate X (Cotton 2026b). This is where conformal prediction sits, as a weak entry; §4 develops the comparison.

The `z2~`/`z3~` copula streams: proper elicitation, distorted metric. Given correct margins, folding two percentiles onto one axis and pricing the folded law elicits the copula, and for the log score the folding is a fixed bijection that changes nothing. The trouble is the *choice* of fold. The Morton z-curve is not nearness-preserving: two joint outcomes that are close on the plane can be far apart on the curve. A nearest-the-pin pool on the folded scalar therefore prices a metric that the curve, not the problem, chose, and a contributor is rewarded for placing mass near the truth *on the curve*. The elicitation is valid; the settlement metric is an artifact. The principled high-dimensional route scores random one-dimensional projections rather than one space-filling curve (Cotton 2026c), averaging a defensible metric instead of fixing an arbitrary one.

The stacked lottery: residual stages composed. The stacked lottery composed the monotone maps of competing algorithms, each remapping the running forecast toward the outcome. This is the residual operator applied in sequence, and it has the same standing as `z1~`: every stage is a proper residual elicitation that pays for the edge it adds, and the composition is a chain of them. Like `z1~`, it rewards sharper prediction at each stage.

The verdict. The chains were the right idea and mostly the right mechanism. Where they settled on an exogenous outcome and scored a residual, they were proper (the residual and copula elicitations). The base pool would have leaked an edge of order the bandwidth, but an accidental jitter introduced for discrete outcomes stood in for the settlement correction and kept it roughly honest. Where they folded a joint onto a space-filling curve, the elicitation held but the implied metric did not.

4 Conformal prediction, nested and embarrassed

Conformal prediction can be read as a degenerate case of a two-stage solicitation: a base forecast, then a residual market in which competition has been restricted, probably unwisely, to a single unconditional entry. This can be seen in two ways.

Forecasters come in two types, and the split has nothing to do with which contest they enter. An *unconditional* forecaster submits a single distribution and stands by it whatever the input; a *conditional* forecaster submits one that moves with the covariates she sees. In any log-wealth (Kelly) pool that settles on an outcome, base or derived, the best conditional entrant out-earns the best unconditional one by exactly the conditional information the covariates carry, $I(R;X)$ (Cotton 2026b); under a general proper score there is still a gap, but it is not this mutual information.

Split-conformal prediction is an unconditional recipe. It forms one residual law from the calibration data and applies it at every input, reading that marginal off the residuals rather than assuming any shape; as a forecaster it simply does not condition. So conformal prediction is the residual market of §2 with the field restricted to unconditional entries. A market would let that entry compete and pay it only while no conditioning rival beats it; conformal instead declares it the answer in advance, forfeiting $I(R;X)$ to anyone who conditions. Adaptive conformal variants that let the law depend on the input are conditional entries, which is to say they are forecasting.

The gap does not close even if the conformists are right. Suppose only they enter the base contest and the base forecast is calibrated, so the z-scores are marginally standard normal. A conditional participant still profits against the marginal submitter, at the same rate: marginal correctness is not conditional correctness, and the difference is the whole downstream game.

5 Related work

Prediction markets have long been read as learning algorithms; the chain here differs from the main lines in one structural choice, that it runs in series.

Storkey's machine learning markets (Storkey 2011) price a single market in which algorithmic agents trade to a simultaneous equilibrium: the ensemble is the clearing price, and the agents' utilities pick out the combination rule, log utility giving a wealth-weighted mixture of experts and an exponential utility a product of experts, the two pools of the companion algebra (Cotton 2026a). The chain trades that simultaneity for modularity. Each stage settles before the next reads its output, which avoids pricing a joint state space all at once but gives up a single global equilibrium and makes the payout depend on the order of the stages.

Abernethy and Frongillo's crowdsourced learning mechanism (Abernethy and Frongillo 2011) publishes one hypothesis and pays each participant in proportion to how much their update improves it, the reward a Bregman divergence and the market a cost function $C(q) = \sup_p \langle p, q \rangle - R(p)$, the same conjugate object as the market makers here. A residual stage rewards marginal improvement in the same spirit, but on the probability integral transform rather than a shared parameter: when the upstream forecast is calibrated the downstream market faces a uniform residual, isolating the conditional inefficiency, and the message is a full distribution, a sample cloud, or a copula rather than a point.

Falconer, Kazempour and Pinson's regression markets (Falconer et al. 2024) pay for shared features by their Shapley value, fair across all coalitions but $O(2^n)$ to compute. The chain sidesteps the combinatorics by fixing an order: a participant at stage k is scored only on what stage $k-1$ left, path-dependent but settleable in real time. The two are complements, Shapley for a fair one-shot split and the chain for streaming settlement.

6 Open problems

1. *Conservation of edge.* Wealth conservation across a chain is automatic and is not the invariant; the edge a truthful participant holds over the price, $D_G(q, \pi)$, can grow or shrink through an interface stage. For the log score the edge is $\mathrm{KL}(q \| \pi)$ and the data-processing inequality caps it (Csiszár 1967); no such inequality holds for a general Bregman edge — coarsening can *raise* the Brier edge. Open: characterize the generators and channels for which an interface cannot increase the edge.
2. *Residual markets.* A chain of residual markets is stagewise boosting with wealth as the learning rate (Proposition 2). Open: who funds each residual pot, whether a forecaster free to enter several stages withholds upstream to sell downstream, and whether the iteration converges to the conditional law. Split-conformal prediction is the one-participant degenerate case, its rent the discarded information $I(R; X)$ (Cotton 2026b).
3. *Copula settlement.* The `z2~`/`z3~` streams show that a copula elicits properly but its settlement metric is the fold's, not the problem's. Open: the right embedding for $d \geq 2$, the space-filling curves as deployed against the random projections of the point-cloud paper

(Cotton 2026c), and the equilibrium when the same participants trade margin and copula stages.

7 Conclusion

The chain treats calibration, dependence, and residual edge as separate objects, each with its own market. A base pool elicits the level, a z-stream prices its calibration, a rank-settled stage prices the copula, and a residual stage prices whatever conditional structure the level left behind. What a single distributional contest folds into one hard-to-price object, the chain lets specialised participants earn against one piece at a time. The cost is the loss of a simultaneous equilibrium and a dependence on the order of the stages; the gain is that a hard, repeated forecast can be assembled, and paid for, one standardised piece at a time, which was the point of a supply chain to begin with.

References

Abernethy, Jacob D., and Rafael M. Frongillo. 2011. "A Collaborative Mechanism for Crowdsourcing Prediction Problems." *Advances in Neural Information Processing Systems (NeurIPS)* 24. https://arxiv.org/abs/1111.2664.

Cotton, Peter. 2019. *Self-Organizing Supply Chains for Micro-Prediction: Present and Future Uses of the ROAR Protocol*. https://arxiv.org/abs/1907.07514.

Cotton, Peter. 2020a. *A Call for Contributions to a Copula Contest*. https://www.linkedin.com/pulse/call-contributions-copula-contest-where-carefully-can-cotton-phd/.

Cotton, Peter. 2020b. *The Lottery Paradox: A New Use*. https://www.slideshare.net/slideshow/lottery-paradox-csaildec2020/242173597.

Cotton, Peter. 2022. *Microprediction: Building an Open AI Network*. MIT Press.

Cotton, Peter. 2024a. *A New Generative AI Competition in Eleven Dimensions*. https://microprediction.medium.com/a-new-generative-model-competition-to-test-the-old-against-the-new-189d3d460669.

Cotton, Peter. 2024b. *Density: Conventions for Densities Used in Various Games*. https://github.com/microprediction/density.

Cotton, Peter. 2024c. *Monteprediction: A Weekly Monte Carlo Competition in Eleven Dimensions*. https://www.monteprediction.com.

Cotton, Peter. 2026a. *An Algebra of Prediction-Rewarding Mechanisms*. https://mechanisms.microprediction.org/papers/composition-and-the-algebra-of-mechanisms.html.

Cotton, Peter. 2026b. *Betting Against a Conformal Predictor: A Parimutuel Account of the Information Gap*. https://conformalprediction.net/papers/parimutuel/.

Cotton, Peter. 2026c. *Scoring Point-Cloud Distributional Submissions*. https://mechanisms.microprediction.org/papers/scoring-point-cloud-distributional-submissions.html.

CrunchDAO. 2024. *The MidOne Contest*. https://mid-one.crunchdao.com/.

Csiszár, Imre. 1967. "Information-Type Measures of Difference of Probability Distributions and Indirect Observations." *Studia Scientiarum Mathematicarum Hungarica* 2: 299–318.

Falconer, Thomas, Jalal Kazempour, and Pierre Pinson. 2024. "Bayesian Regression Markets." *Journal of Machine Learning Research* 25. https://arxiv.org/abs/2310.14992.

Friedman, Jerome H. 2001. "Greedy Function Approximation: A Gradient Boosting Machine." *The Annals of Statistics* 29 (5): 1189–232. https://doi.org/10.1214/aos/1013203451.

Hanson, Robin, and Ryan Oprea. 2009. "A Manipulator Can Aid Prediction Market Accuracy." *Economica* 76 (302): 304–14. https://doi.org/10.1111/j.1468-0335.2008.00734.x.

Harville, David A. 1973. "Assigning Probabilities to the Outcomes of Multi-Entry Competitions." *Journal of the American Statistical Association* 68 (342): 312–16. https://doi.org/10.1080/01621459.1973.10482425.

Hausch, Donald B., William T. Ziemba, and Mark Rubinstein. 1981. "Efficiency of the Market for Racetrack Betting." *Management Science* 27 (12): 1435–52. https://doi.org/10.1287/mnsc.27.12.1435.

Jarrow, Robert A. 1994. "Derivative Security Markets, Market Manipulation, and Option Pricing Theory." *Journal of Financial and Quantitative Analysis* 29 (2): 241–61. https://doi.org/10.2307/2331224.

Jha, Akshaya, and Frank A. Wolak. 2023. "Can Forward Commodity Markets Improve Spot Market Performance? Evidence from Wholesale Electricity." *American Economic Journal: Economic Policy* 15 (2): 292–330. https://doi.org/10.1257/pol.20200234.

Kumar, Praveen, and Duane J. Seppi. 1992. "Futures Manipulation with 'Cash Settlement'." *The Journal of Finance* 47 (4): 1485–502. https://doi.org/10.1111/j.1540-6261.1992.tb04665.x.

Mason, Llew, Jonathan Baxter, Peter L. Bartlett, and Marcus Frean. 1999. "Boosting Algorithms as Gradient Descent." *Advances in Neural Information Processing Systems (NeurIPS)*, 512–18.

Ostrovsky, Michael. 2012. "Information Aggregation in Dynamic Markets with Strategic Traders." *Econometrica* 80 (6): 2595–647. https://doi.org/10.3982/ECTA8479.

Sklar, Abe. 1959. "Fonctions de répartition à n Dimensions Et Leurs Marges." *Publications de l'Institut de Statistique de l'Université de Paris* 8: 229–31.

Smith, Adam. 1776. *An Inquiry into the Nature and Causes of the Wealth of Nations*. W. Strahan; T.

Cadell.

Storkey, Amos J. 2011. "Machine Learning Markets." *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)*, PMLR, vol. 15: 716–24. https://arxiv.org/abs/1106.4509.