A computational model of feature formation, event prediction, and attention switching

Eman Awad and **Fintan Costello**
School of Computer Science,
University College Dublin,
Belfield, Dublin 6, (eman.awad@ucdconnect.ie,fintan.costello@ucd.ie)

Abstract

In this paper we present a model of three central aspects of probabilistic cognition: event prediction, feature formation, and attention allocation. While most models of probabilistic reasoning take a parameter estimation and error minimisation approach (sometimes referred to as 'predictive coding', and often described in terms of Bayesian updating), our model takes a contrasting frequentist hypothesis-testing approach. This choice is motivated by a series of recent results suggesting that people's probabilistic reasoning follows frequentist probability theory. In simulation tests we demonstrate that this frequentist model, in which predictive features are formed by a process of null hypothesis significance testing, can give a successful account of event prediction and attentional switching behaviour.

Introduction

There are, broadly speaking, two approaches to statistical reasoning: a 'parameter estimation' approach (associated primarily with Bayesian statistics), where some form of generative model is used to predict data and the estimation process involves adjusting parameters of this model so as to reduce errors in prediction; and a 'hypothesis testing' approach (associated primarily with frequentist statistics) where a decision is made to reject a hypothesis (that is, to reject a possible generative model) when the probability of the observed data under that model is less than some significance level. Most current models of probabilistic cognition, learning and attention take the parameter estimation and error minimisation approach, sometimes referred to as 'predictive coding'; this approach is naturally described in terms of Bayesian priors (values of generative model parameters) which are 'updated' by experience, to produce more accurate posterior estimates of those parameters (see e.g. Clark, 2013; Griffiths and Tenenbaum, 2006; Tenenbaum et al., 2011; Miller et al., 1995).

In this paper we present a model of probabilistic cognition based on frequentist hypothesis-testing rather than parameter estimation and error minimisation. We apply this model to the processes of probabilistic learning, feature formation, event prediction, and attention. There are three motivations for this frequentist hypothesis-testing approach to probabilistic cognition. First, the contrasting parameter estimation and hypothesis-testing approaches to statistical reasoning are known to have different strengths and weaknesses: modelling probabilistic cognition via frequentist hypothesis-testing is worthwhile because it allows us to see this type of cognition in a new light.

Second, the hypothesis-testing approach applies very naturally to one core aspect of probabilistic cognition; that of decision making. Decision making is central to feature formation (given observed pattern of co-occurrence between events, how do we decide whether to treat that pattern as representing a single complex event, and so form a feature representing that event?), event prediction (given estimated probabilities of various future events or outcomes, how do we decide which event to predict?) and attention (given multiple sources of information, how do we decide whether to direct our attention to one source rather than another?) The frequentist hypothesis-testing approach was specifically developed to guide decision-making on the basis of data (see e.g. Fisher, 1937), and so provides a natural normative framework for modelling decision making in prediction, feature formation and attention allocation.

Finally, this model is motivated by recent evidence suggesting that people's probabilistic reasoning processes follow the requirements of frequentist probability theory (Costello and Watts, 2018a, 2016), and that a range of well-known biases in probabilistic reasoning can be explained as a consequence of regression produced by random variation or noise in normatively correct frequentist reasoning (Costello and Watts, 2014, 2018b). The model described here represents a computational implementation of this account; in this model random variation arises simply as a consequence of sampling.

We present this model incrementally, focusing first on prediction, feature formation and probabilistic learning for a single 'stream' of input (that is, with fixed attention). We then generalise to learning, feature formation and prediction across multiple simultaneous streams of input (where attention moves from stream to stream). We test the frequentist approach by comparing the effectiveness of an attention switching mechanism derived from frequentist hypothesis testing against the effectiveness of a switching mechanism based on error minimisation (as used in predictive coding), and against a random switching baseline.

The model

At an abstract level, temporal prediction involves taking a temporally ordered stream of categorical events or labels, such as

$$A,B,S,A,B,S,A,-,S,A,-,S,-,A,B,S,A,B,A,B \qquad (1)$$

and predicting the next event in the stream. Our model predicts future events in such sequences by constructing features from observation of a given stream, identifying features which allow statistically reliable predictions, and then combining these 'predictive features' to give an overall predicted probability for the next event in the stream.

Each feature in our model consists of an antecedent event A, a consequent event S, and a time interval t between them. Each feature also holds two counts: k, a count of the number of times A has been followed, after time t, by S; and n, a count of the number of times A has been followed, after time t, by any event. Finally, each feature holds a conditional probability $P(S|A) = k/n$, representing the probability of seeing the consequent S at time t after the occurrence of the antecdent A.

The antecedent A in a given feature may be a single event (e.g. the label A as a predictor of the next event, in our example in (1)), or may be a combination of events occurring over time (e.g the consecutive labels A,B as a predictor of the next event). Our model stores, in 'Short Term Memory' (*STM*), the N most recent events in the stream. Our model stores, in 'Long Term Memory' (*LTM*), a large number of simple or complex features that have been observed, with some features marked as 'reliable', meaning that there is statistically significant evidence supporting the relationship between antecedent A and consequent S in that feature. These reliable features are used to make predictions about the next event in the stream. The model has two free parameters: N, the size of short term memory, which we set by default at 4, and c, the significance criterion, which we set by default at $c = 0.05$.

Reliable features and prediction

To decide whether a given feature describes a statistically reliable relationship between antecedent A and consequent S, our model follows the hypothesis-testing approach of standard frequentist probability theory. We consider two possible cases: one where the antecedent event is a single 'atomic' event; and one where the antecedent is a complex or composite event, made up of multiple subevents.

In the case of a single event as antecedent A, we have two hypotheses: a null hypothesis (that there is no relationship between A and S; under this hypothesis the probability getting S after A is simply the base probability of event S, $P(S)$) and an alternative hypothesis (that there is a reliable relationship between A and S; under this hypothesis the probability seeing S after A is given by $P(S|A)$). The probability of obtaining k instances of S after A in a sample of n occurrences of A, assuming that $P(A) = p$, is given by the binomial

$$Bin(k,x,p) = \binom{n}{k} p^k (1-p)^{n-k} \tag{2}$$

This means that if $Bin(k,n,P(S)) < c$, for some critical significance level c, then we can reject the null hypothesis that $P(A) = P(S)$, and can instead accept the alternative hypothesis, that there is a reliable predictive relationship between A and S. When $Bin(k,n,P(S)) < c$ the model thus marks the feature linking A with S after time t as a statistically reliable predictive feature.

We now consider the situation where we have a complex event made up of sub-events A and B (each of which may itself be made up of further subevents), and where this complex event *AthenB* is itself an antecedent of our consequent S. Here we take k to represent the number of times consequent S has occurred at time t after antecedent *AthenB* in the observed time series, and n to represent the number of times any event at all has occurred at time t after antecedent *AthenB* in the series. In this situation we test against three possible 'null hypotheses': that $P(S|AthenB) = P(S)$, as before; that $P(S) = P(S|A)$ (that the probability getting S after *AthenB* is simply the probability of getting S after A, $P(S|A)$); and that $P(S) = P(S|B)$ (that the probability getting S after *AthenB* is simply the probability of getting S after A, $P(S|B)$). These three 'null hypotheses' are tested using the binomial as before. If all three tests are significant, the model concludes that the complex feature *AthenB* is itself a distinct, statistically reliable predictor of the occurrence of S: given that *AthenB* has occurred, the probability of S is given by $P(S|AthenB)$ (rather than by $P(S), P(S|A)$ or $P(S|B)$). By contrast, if there is no additional relationship between the feature *AthenB* and S beyond that given by A, B, and the base rate $P(S)$ then we can say that A and B (or some combination of their subevents) are independent predictors of the consequence S.

Using this hypothesis-testing procedure our model identifies, for a given sequence of events, the set of independent, statistically reliable, features occurring in that sequence which predict a given event S at the next timestep. Following standard frequentist probability theory the overall predicted probability of S is calculated by 'ORing' these independent predictions

$$Pr(S|A_1,\ldots,A_n) = 1 - \prod_{i=1..n} (1 - P(S|A_i)) \tag{3}$$

to give an overall predicted probability for S occurring next in the sequence.

An example

We can give an example of the model's operation for our example sequence in (1). In that series there are 20 events in total, of which 5 are S (so $P(S) = 0.25$). There are 3 occurrences of B followed one step later by S, and 1 occurrence of B followed one step later by a different event. To test the hypothesis that the occurrence of B predicts the occurrence of S at the next time step, we calculate the probability of seeing 4 occurrences of B, 3 of which are followed by S, if S was occurring at its base rate probability of 0.25: $Bin(3,4,0.25) = 0.0469$. Since this probability is less than our significance criterion of $c = 0.05$ we conclude that there is a statistically reliable relationship between B and S. Similarly, there are 5 occurrences of A followed two steps later by S, and 1 occurrence of A followed two steps later by a different event. To test the hypothesis that the occurrence of A predicts the occurrence of S

two steps later, we calculate the probability of seeing 6 occurrences of A, 5 of which are followed by S, if S was occurring at its base rate probability of 0.25: $Bin(5,6,0.25) = 0.0004$. This probability is also less than our significance criterion of 0.05 and we conclude that there is a statistically reliable relationship between A and S.

We also consider the complex predictive feature *AthenB*. There are 3 occurrences of *AthenB* followed one step later by S, and 1 occurrence of *AthenB* followed one step later by a different event. Since we've seen 4 occurrences of B, 3 of which were immediately followed by S, we estimate $P(S|B) = 3/4 = 0.75$. Testing against the hypothesis that the observed relationship between *AthenB* and S is explained simply be the presence of B we calculate the binomial probability $Bin(3,4,0.75) = 0.42$, and we see that there is no evidence for an additional relationship between the *AthenB* and S beyond that given by B. Since we've seen 5 occurrences of A, 4 of which were immediately followed by S, we estimate $P(S|A) = 4/5 = 0.8$, and since $Bin(3,4,0.8) = 0.41$ we similarly see that there is no evidence for an additional relationship between the *AthenB* and S beyond that given by A. We can thus conclude that the complex event *AthenB* is not a reliable predictor of S; instead, the two simple events A and B are independent predictors of the occurrence of S, with A predicting S after 2 steps with $P(S|A) = 0.8$, and B predicting S after 1 step with $P(S|B) = 0.75$. If events A, B have just occurred, the probability of S occurring next is obtained by ORing the predictions of these two statistically reliable features, giving

$$P(S) = 1 - (1 - P(S|A))(1 - P(S|B)) = 1 - 0.2 \times 0.25 = 0.95$$

as the predicted probability of S occurring at the next timestep in the stream shown in (1).

Switching between multiple streams

In this section we apply this model to feature formation and prediction across multiple different streams of input. We assume a single Long Term Memory (*LTM*), as before. To deal with multiple streams of input we assume multiple separate *STM* stores, one for each stream, and with each *STM* storing the last N events that have occurred in that stream. The model uses the statistically reliable features in *LTM* to calculate predicted probabilities for the next event in each input stream. The predicted next event for input stream i is calculated by finding statistically reliable predictive features in *LTM* whose antecedent event has occurred in STM_i (that is, in the store of recent events from stream i), and then combining the predictions from those features as described above.

Prediction, for each stream, happens in parallel and is computationally cheap (there are typically very few statistically reliable predictive features whose antecedents are present in a given *STM*). Learning and feature formation, however, are computationally 'expensive' and so take place only for one particular stream: the stream that is the current focus of attention. The model forms new features, updates antecedent and consequent occurrence counts, and identifies statistically reliable predictive features just as before, but only for this focal stream.

As the overall goal of the model is to accurately predict its environment (to accurately predict event occurrence in all streams), the model must occasionally switch its focus of attention from one stream to another. Attentional switching is a form of decision making: the model must decide to switch attention away from one stream of input (and so cease any predictive learning from events in that stream) and towards another stream of input (so beginning the process of learning from events in that new stream). The overall goal of the model is to form statistically reliable predictive features; satisfaction of this goal requires a decision process where attention is switched towards streams where statistically reliable predictive features are more likely to be formed, and away from streams where such reliable features are less likely to be formed. As before, frequentist hypothesis testing gives a natural and normatively correct way to make such switching decisions.

Suppose we are considering forming a reliable feature *ApredictsS*, and have observed k instances of A followed by S, out of n occurrences of A overall. This feature will be judged reliable when the observed pattern of co-occurrence between A and S has a low probability of occurrence under the null hypothesis that A does not predict S (when $bin(k,n,P(S)) < c$). This means that the lower the value of this binomial expression $bin(k,n,P(S))$, the more likely it is that the null hypothesis is false and the there is some reliable relationship between A and S. In other words, if we have we have some feature for which $bin(k,n,p) > c$ (some feature which is not yet reliable), then the probability that this feature will become reliable is proportional to $1 - bin(k,n,p)$. More generally, if a given stream i contains a number of not-yet-reliable features with counts k_j and n_j and null hypothesis values p_j, then the overall probability of forming a reliable feature in that stream is obtained by ORing the individual probabilities of each of these features becoming reliable, as given in the expression

$$F(i) = 1 - \sum_{\substack{\text{j= not yet} \\ \text{reliable} \\ \text{feature in} \\ \text{stream i}}} bin(k_j, n_j, p_j) \qquad (4)$$

The greater the value of this expression $F(i)$ for a given stream i, the greater the probability that switching attention to that stream will lead to the formation of a new reliable predictive feature. The model uses these values $F(i)$ to guide switching; at each timestep the model calculates $F(i)$, the probability of constructing a new reliable feature, for all streams i including the current stream x, and identifies the stream *max* with the highest value. If $F(max) - F(x) > s$, where s is a switching decision criterion (if the chance of forming a new feature in stream *max* is s greater than the chance of forming a new feature in the current stream) then the model switches attention to stream *max*; otherwise attention remains in the current stream.

Testing the Model

We test our model via Markov Chain Monte Carlo simulation, as follows.

Any process producing a series of events can be represented by an nth order Markov chain (for some value of n). Such chains thus represent realistic generative models of sequential event occurrence. For each stream of information in our simulation, we construct a generative n-th Markov chain with $m = 4$ (4 distinct categorical events) and $n = 4$ (the current state consists of the last 4 events). There are $4^4 = 256$ distinct states in this chain, each with 4 transition probabilities. For each state these 4 transition probabilities are assigned random values, normalised so their sum for that state equals 1. Each stream thus represents a (randomly constructed) Markov chain process.

We use the Markov chain to generate a large sequence of categorical events from that stream. We first pick an 4 initial events at random, representing the initial state of the Markov Chain. We identify the 4 transition probabilities associated with that state, and choose one transition (one new event) at random, proportional to its transition probability from the current state in the Markov Chain. The selected event is added to our sequence of generated events. The new state of our Markov model now consists of the 4 most recent events (three previous events and the event that was just added to the series), and the cycle repeats.

The events for each stream are fed in parallel to our model, which forms statistically reliable predictive features, makes predictions for the next event to occur at each time step in each stream, and switches attention between streams as described above.

After an initial training phase we continue running the model and the Markov Chain generators for an additional test phase. We gather, at each time step of this phase, the model's predicted probability for each event in each stream, and whether or not that event actually occurred. We assess the model by gathering together all cases where the model predicts that an event will occur with probability in some range R. If the model is accurate, the proportion of those predicted events that did actually occur should be in or near the range R. For example: we gather together all cases where the model predicted some event with probability in the range $0.1 - 0.15$. If the model's predictions are accurate, then the predicted event should have actually occurred around $10\% - 15\%$ of the time; the probability (or proportion) of actual occurrence of the predicted event should be close to the range $0.1 - 0.15$.

Test 1: Learning from a single stream

Table 1 shows results obtained when running the model with a single stream of input (no switching between streams), for a 5000 timestep training phase (during which the model formed predictive features) followed by a 5000 timestep test phase (during which we gathered the model's predicted probabilities for the next event, at each timestep). This table gives the proportion of times the model's predicted event occurred, for prediction ranges from $0.05 - 0.10$ to $0.55 - 0.60$. As the table demonstrates, the model's predicted probabilities corresponded closely with the actual probability (or proportion) of occurrence of the predicted event.

Table 1: This table shows the number of times our model predicted that an event would occur with a probability that fell into a given range **R** (column 2), and of those predictions, the number of times when the predicted event actually occurred (column 3). If the model is making accurate predictions, the proportion of occurrence of the predicted event (the observed probability, column 4) should follow the range value R. The two values are highly correlated ($r = 0.99$) indicating that the model is predicting event probabilities accurately.

Predicted probability range **R**	Number of predictions in **R**	Predicted event occurred	Observed probability of predicted event
0.05 - 0.10	1393	193	0.14
0.10 - 0.15	2268	401	0.18
0.15 - 0.20	2600	516	0.19
0.20 - 0.25	3016	6463	0.21
0.25 - 0.30	3901	1094	0.28
0.30 - 0.35	3341	1005	0.3
0.35 - 0.40	1806	597	0.33
0.40 - 0.45	788	272	0.35
0.45 - 0.50	307	117	0.38
0.50 - 0.55	171	64	0.47
correlation with probability range **R**			0.99

As Table 1 also demonstrates, there is regression in the model's predictions: for low predicted probability ranges, observed event probabilities tend to be significantly higher than the probability range, while for high predicted probability ranges, observed event probabilities tend to be significantly lower than the probability range. This pattern of regression in turn implies that the models predicted probabilities are regressive towards the center of the probability scale, relative to true event probabilities Erev et al. (1994). This pattern of regression is just as assumed in Costello & Watts frequentist account of probabilistic reasoning (Costello and Watts, 2014, 2016, 2018a,b). This model thus provides a mechanistic implementation of that account, in which regression arises as a consequence of random sampling variation.

Test 2: Learning from a multiple streams

To test the hypothesis-testing model of switching given above, we test the model in the same Random Markov chain regime, but with 5 parallel streams of input, each with its own randomly initialised Markov Chain generator. Specifically, we compare learning under this model against learning under random switching, and learning under an alternative 'predic-

Table 2: Average observed probability of occurrence of predicted event for prediction range *R*, across all streams. Observed probabilities are calculated as in Table 1. Data is given for each switching mechanism, running the model for 5000 times/steps in each case. Both the 'Switch to max error' and the 'switch to form reliable features' switching methods gave predictions closely correlated with observed probability.

Range **R**	Observed probability of predicted event for predictions in range **R** (by Switching method)		
	Random switching	Switch to max. error	Form reliable features
0.05 - 0.10	0.17	0.00	0.14
0.10 - 0.15	0.18	0.16	0.17
0.15 - 0.20	0.17	0.17	0.19
0.20 - 0.25	0.17	0.19	0.21
0.25 - 0.30	0.2	0.22	0.25
0.30 - 0.35	0.34	0.32	0.3
0.35 - 0.40	0.41	0.39	0.34
0.40 - 0.45	0.48	0.45	0.39
0.45 - 0.50	0.56	0.49	0.39
0.50 - 0.55	0.49	0.61	0.42
correlation	0.90	0.98	0.99

tive coding' model of attentional switching, where we switch attention to the stream where errors in event prediction are highest.

Random switching As a baseline for comparison, we run the model with a fixed-length random-choice method for switching between streams. Under this switching method, the model will remain in a certain stream for 100 timesteps at a stretch; after each sequence of 100 timesteps has passed, the model will switch to a randomly selected other stream. Given that the learning model performs well in learning to predict events in a single stream, we expect that the model will perform relatively well in predicting events across multiple streams under this random switching regime.

Switching to minimise predictive error As an alternative for comparison, we run the model with a switching method designed to minimise predictive error. In the predictive coding view, a learning model makes predictions which are compared with outcomes: attention is driven towards locations where those predictions are incorrect (and so more learning is required) and away from locations where predictions are accurate (and so less learning is needed). In our model, predictive error in a given stream at a given time is simply equal to the predicted probability of the event that actually occurred in that time: if S is the event that actually occurs and the model's prediction probability for S was high, then there is little predictive error; if S occurs and the model's prediction probability of S was low, there is significant predictive error.

Table 3: Average correlation between observed and predicted event probability, obtained after learning with each switching method, for runs of different length (500,1000,5000,10000 and 50000 times/steps). Both the 'Switch to max error' and the 'switch to form reliable features' switching methods gave predictions that were more closely correlated with observed even occurrence rates, with the 'switch to form reliable features' approach giving the highest average correlation between observed and predicted probability.

Run size	Correlation between observed and predicted probability (by Switching method)		
	Random switching	Switch to max. error	Form reliable features
500	0.85	0.96	0.88
1000	0.89	0.97	0.96
5000	0.9	0.98	0.99
10000	0.9	0.98	0.99
50000	0.9	0.96	0.99

To implement a switching mechanism based on predictive error, we give a method which calculates, for each stream i, the average predicted probability of the last N events that occurred in this stream. The lower this average, the more prediction error in stream i. Letting $G(i)$ be equal to 1 minus the average prediction probability for stream i, the model uses a decision criterion s to guide switching; at each timestep the model calculates $G(i)$ for all streams i including the current stream x, and identifies the stream *max* with the highest value. If $G_{max} - G(x) > s$ (if prediction error in stream *max* is s greater than that in the current stream) then the model switches attention to stream *max*; otherwise attention remains in the current stream.

Results

We ran the model separately with each of the three different switching mechanisms described above, and with different number of training and test timesteps (500, 1000, 5000, 10000 and 50000 timesteps: in each case the training phase was the same length as the test phase). As before, we grouped the model's predictive probabilities into a series of 'buckets' or ranges **R** (so one bucket would hold all cases where the model predicted some event with a probability between 0.05 and 0.1, another would hold all cases where the model predicted some event with a probability between 0.1 and 0.15, and so on). For each bucket we counted the number of times the predicted event actually occurred. If the model is accurate, the proportion of those predicted events that did actually occur (the observed probability of occurrence of predicted events), for a given range R should be in or near the range R (the predicted probability for those events).

Table 2 shows the results of this analysis of model prediction for the 5000 timestep run with each of the three switching methods. This table gives the observed probability of events whose predicted probability fell in range R, in runs of the model with each of the 3 possible switching methods. The observed probabilities shown here are averages across predictions made in all 5 parallel streams of input, in a single run of the model (observed probabilities in each stream closely follow the pattern seen here, and closely follow the pattern seen in Table 1). As this table shows, there was a reliable correlation between the range in which the model predicted an event will occur and the actual rate or ratio of occurrence of that event, for all switching methods. This is expected: as we saw in the earlier 'single-stream' simulation tests, the model does well in learning to predict events accurately from observed event sequences, and so we would expect the model to learn to predict all streams relatively well no matter what attentional switching mechanism was being used.

The table also shows that both the 'switch to form reliable features' and the 'switch to maximum error' methods give results that matched observed probabilities much more closely than those given by the 'random switching' mechanism. These results demonstrate the contribution that effective attentional switching can make to prediction accuracy.

Table 3 shows the correlation between model predicted probabilities and observed event probabilities for each switching method, across increasing training and test time. As this table shows, correlation between predicted and observed probabilities increased with learning to some degree for all switching methods, but increased to very high correlation values for both the 'switch to max error' and the 'switch to form reliable features' methods. Taken together, these results demonstrate that this hypothesis-testing model forms features which reliably predict future events, and switches attention in a way that maximises formation of such features.

Conclusions

We have described a computational model of prediction, feature formation, and attentional switching. This model is interesting because it is based on the frequentist, hypothesis-testing approach to statistical reasoning, as opposed to the parameter-estimation approach currently popular in models of these cognitive processes. This model represents a computational implementation of a general account of probabilistic reasoning, also based on frequentist probability (Costello and Watts, 2014, 2016, 2018a). That account sees human probabilistic reasoning as being based on normatively correct processes, but subject to random variation or 'noise': that noise has systematic regressive effects, producing a range of biases in people's probabilistic judgement. The computational model implemented here demonstrates just the pattern of regression assumed in that more general account, and so inherits its account for those biases.

The frequentist, hypothesis-testing approach described here may usefully address two problems with the standard parameter-estimation approach to probabilistic prediction. These problems arise because the Bayesian approach to learning and prediction often require the specification of initial priors, in two separate ways. First, such models require initial assumptions as to the form of the generative model being used to predict data (assumptions which specify which features are 'available' for use in prediction, for example, and which features are not). The hypothesis-testing approach described here in some ways avoids this requirement, by providing a mechanism whereby predictive features are 'built' out of observed event. Second, such models require assumptions as to the initial values of parameters in that generative model (assumptions about the initial, prior, probability distribution associated with features in the generative model). The hypothesis-testing approach described here avoids this requirement also, because features are not identified as 'reliable' (and so do not contribute to predictions) until the events making up those features have been repeatedly observed (that is, until any initial 'prior' has been made irrelevant by repeated experience with the events in question). These points suggest that an integrated approach, combining the parameter-estimation and the hypothesis-testing perspectives, may prove insightful. Understanding the interplay between hypothesis testing and parameter estimation in human probabilistic reasoning is an important aim for future work.

References

Clark, A. (2013). Whatever next? predictive brains, situated agents, and the future of cognitive science. *Behavioral and brain sciences*, 36(3):181–204.

Costello, F. and Watts, P. (2014). Surprisingly rational: probability theory plus noise explains biases in judgment. *Psychological review*, 121(3):463.

Costello, F. and Watts, P. (2016). Peoples conditional probability judgments follow probability theory (plus noise). *Cognitive Psychology*, 89:106–133.

Costello, F. and Watts, P. (2018a). Invariants in probabilistic reasoning. *Cognitive psychology*, 100:1–16.

Costello, F. and Watts, P. (2018b). Probability theory plus noise: Descriptive estimation and inferential judgment. *Topics in cognitive science*, 10(1):192–208.

Erev, I., Wallsten, T. S., and Budescu, D. V. (1994). Simultaneous over- and underconfidence: The role of error in judgment processes. *Psychological Review*, 101(3):519–527.

Fisher, R. A. (1937). *The design of experiments*. Oliver And Boyd; Edinburgh; London.

Griffiths, T. L. and Tenenbaum, J. B. (2006). Optimal predictions in everyday cognition. *Psychological science*, 17(9):767–773.

Miller, R. R., Barnet, R. C., and Grahame, N. J. (1995). Assessment of the rescorla-wagner model. *Psychological bulletin*, 117(3):363.

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., and Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *science*, 331(6022):1279–1285.