

SafePredict: a meta-algorithm for machine learning to guarantee correctness by refusing occasionally

Mustafa A. Kocak¹, David Ramirez¹, Elza Erkip¹, and Dennis E. Shasha²

¹ NYU Tandon School of Engineering

² NYU's Courant Institute of Mathematical Sciences

Introduction

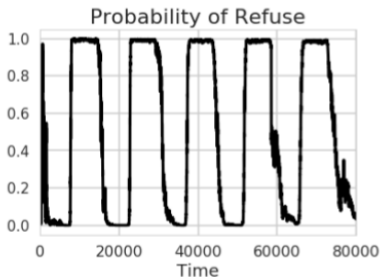
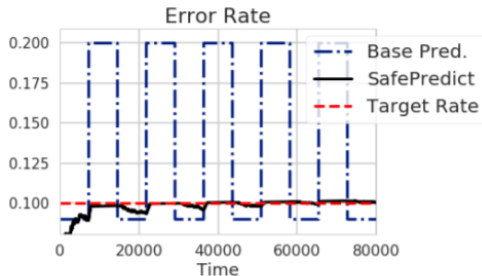
- Machine learning and prediction algorithms are the building blocks of automation and forecasting.
- Reliability is crucial in risk-critical applications.
 - Analytics, risk assessment, credit decisions.
 - Health care, medical diagnosis.
 - Judicial decision making.
- **Basic idea:** Create a meta-algorithm that takes predictions from underlying machine learning algorithms and decides whether to pass them on to higher level applications.
- **Goal:** Achieve *robust correctness guarantees* for the predictions emitted by the meta-algorithm.

What Does It Mean to Refuse?

- The implications of refusing to make a prediction may vary according to the application of interest.
 - do more tests / collect more data
 - request user feedback or ask for a human expert to make the decision.
- Want to refuse seldom while still achieving the error bound.

Novelty and Teaser

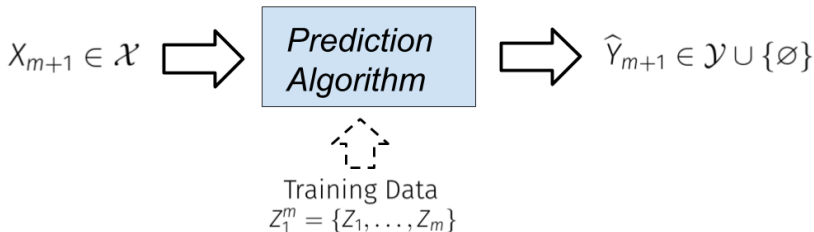
- *SafePredict* achieves a desired error bound without any assumption on the data or the base predictor.
- Tracks the changes in the error rate of the base predictor to avoid refusing too much.



Literature Review

Batch Setup

Data: $Z_i = (X_i, Y_i) \in \mathcal{X} \times \mathcal{Y} \sim \text{i.i.d. } \mathcal{D}$ for all $i = 1, \dots, m + 1$.



Probability of Error (P_e)

$$P\left(\hat{Y}_{m+1} \notin \{Y_{m+1}, \emptyset\} \mid Z_1^m\right)$$

Probability of Refusal (P_r)

$$P\left(\hat{Y}_{m+1} = \emptyset \mid Z_1^m\right)$$

BATCH SETUP GOAL: Minimize $P_e + \kappa P_r$, where κ is the cost of a refusal relative to an error.

Related Work (Batch Setup)

- *Chow, 1970*: Assuming \mathcal{D} is known, the optimal refusal mechanism is:

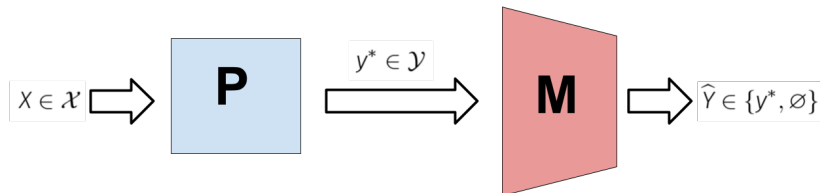
$$\hat{Y}(X) = \begin{cases} y^* & \text{if } P(Y = y^* | X = x) \geq 1 - \kappa \\ \emptyset & \text{otherwise} \end{cases},$$

where $y^* = \arg \max_y P(Y = y^* | X)$ is the MAP predictor.

- For unknown \mathcal{D} , instead minimize $\hat{P}_e + \kappa \hat{P}_r$.
 - *Wegkamp et al., 2006-2008*: Rejection with hinge loss and lasso.
 - *Wiener and El Yaniv, 2010-2012*: Relationship with active learning and selective SVM.
 - *Cortes et al., 2016-2017*: Kernel based methods and boosting.

Refuse Option via Meta-Algorithms

In practice, a meta-algorithm approach is much more common.



Base Predictor P is characterized by a scoring function S :

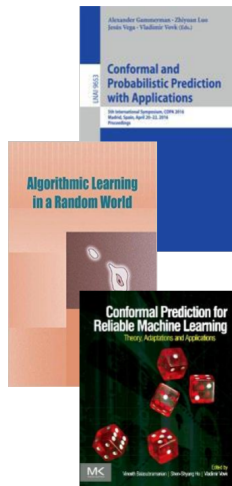
- $S(X, Y)$: How typical/probable/likely is (X, Y) ?
- $y^* = \arg \max_{y \in \mathcal{Y}} S(X, y)$

Meta-algorithm M characterized by τ : $\hat{Y}(X) = \begin{cases} y^* & \text{if } S(X, y^*) \geq \tau \\ \emptyset & \text{otherwise} \end{cases}$.

Conformal Prediction

Conformal Prediction (Vovk et al., 2005):

- Conformity score, $S(x, y)$, measures how well (x, y) conforms with the training data.
 - e.g. distance to the decision boundary, out-of-bag scores, other probability estimates.
- Strong guarantees in terms of coverage, i.e. $P_e \leq \epsilon + o(1)$.
- Probability of refusal is asymptotically minimized if S is consistent.



Probability of error on non-refused data points

- A more practical quantity of interest, probability of error given not refused:

$$P_{e|\bar{r}} := P\left(\widehat{Y}_{m+1} \neq Y_{m+1} \mid \widehat{Y}_{m+1} \neq \emptyset, Z_1^m\right) = \frac{P_e}{1 - P_r}.$$

- There are two main approaches to approach this problem:
 1. **Conjugate Prediction:** For any given scoring function S , calibrate the threshold τ , to guarantee $P_{e|\bar{r}} \leq \epsilon$.
 2. **Probability Calibration:** Fix $\tau = 1 - \epsilon$, and learn a monotonic function F , to calibrate the scoring function S , i.e.

$$F(S(x, y)) \simeq P(Y = y | X = x).$$

Typical methods: *Isotonic* and *Platt's* regression.

Conjugate Prediction - Calibration Step

1. Split the training set as **core training**, Z_1^n , and **calibration**, Z_{n+1}^{n+l} , sets where $n + l = m$.
2. Train the base classifier P on the **core training set**.
3. Choose the smallest threshold τ^* that gives an empirical error rate less than ϵ on the **calibration set**, i.e.

$$\tau^* = \inf \left\{ \tau : \frac{\sum_{i=m+1}^{m+l} 1_{\hat{Y}_i \notin \{Y_i, \emptyset\}}}{\sum_{i=m+1}^{m+l} 1_{\hat{Y}_i \neq \emptyset}} \leq \epsilon \right\}$$

Theorem: At least with probability $1 - \delta$, we get

$$P_{e|\bar{r}} \leq \epsilon + \frac{1}{1 - P_r} \sqrt{\frac{\log(l/\delta)}{2l}}.$$

Empirical Comparison

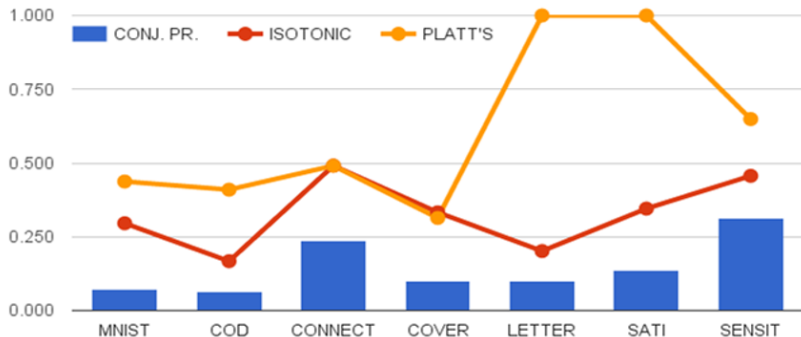
Base Predictor (P) : Random Forest (100 trees).

Scoring Function (S(x,y)) : Fraction of trees that predicts the label of x as y.

Baseline: Train a Random Forest over 75% of the data and test on remaining 25%.

Core/Calibrate/Test Split : 50/25/25

Refuse Rates for Target = Baseline/2



Empirical Comparison

Dataset	Target Error Rate	Conjugate Prediction		Isotonic Regression		Platt's Regression	
		Error Rate	Refuse Rate	Error Rate	Refuse Rate	Error Rate	Refuse Rate
MNIST	0.016	0.017	0.071	0.001	0.296	0.001	0.437
COD	0.009	0.009	0.066	0.004	0.167	0.001	0.410
CONNECT	0.065	0.067	0.239	0.017	0.491	0.017	0.491
COVER	0.024	0.025	0.102	0.004	0.333	0.005	0.314
LETTER	0.005	0.005	0.102	0.000	0.202	0.000	1.000
SATI	0.014	0.013	0.138	0.001	0.346	0.000	1.000
SENSIT	0.082	0.083	0.312	0.062	0.457	0.028	0.649

Probability calibration tends to be too conservative, thus leads to excessive refusals.

Online/Adversarial Setup

- **Online:** First observe x_1, \dots, x_t and y_1, \dots, y_{t-1} , then predict \hat{y}_t .
 - For each $t = 1, \dots, T$:
 - i. Observe x_t .
 - ii. Predict \hat{y}_t .
 - iii. Observe y_t and suffer $l_t \in [0, 1]$.
- **Adversarial:** Assume nothing about the data.
 - Instead assume access to a set of predictors : P_1, P_2, \dots, P_N .

Related Work (Online /Adversarial Setup)

- i. **Realizable Setup:** Assume there exists a perfect predictor in the ensemble.
 - “*Knows What it Knows*” (Li et al., 2008): Minimize the number of refusals without allowing any errors.
 - “*Trading off Mistakes and Don’t-Know Predictions*,” (Sayedi et al, 2010): Allow up to k errors and minimize the refusals.

- ii. **l -bias Assumption:** One of the predictors makes at most l mistakes.
 - “*Extended Littlestone Dimension*” (Zhang et al., 2016): Minimize the refusals while keeping the number of errors below k .

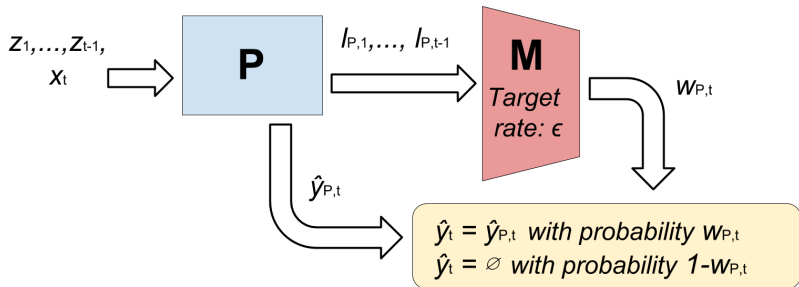
SafePredict

SafePredict is a meta-algorithm for the online setup, which guarantee that the error rate on the non-refused predictions is bounded by a user-specified target rate.

Our error guarantees do not depend on any assumption about the data or the base predictor, but are asymptotic in the number of non-refused predictions.

The number of refusals depends on the quality of the base predictor and can be shown to be small if the base predictor has a low error rate.

Meta-Algorithms in Online Prediction Setup



- Base-algorithm P makes prediction $\hat{y}_{P,t}$ and suffer $l_{P,t} \in [0, 1]$.
- Meta-algorithm M makes a (randomized) decision to refuse (\emptyset) or predict \hat{y}_t , to guarantee a target error rate ϵ .
 - M predicts at time t with probability $w_{P,t}$.

Validity and Efficiency

- We use the following * notation to denote the averages over the randomization of M , i.e.

T^* : Expected number of (non-refused) predictions, $\sum_{t=1}^T W_{P,t}$.

L_T^* : Expected cumulative loss of M , $\sum_{t=1}^T l_{P,t} W_{P,t}$.

Validity

M is valid if $\limsup_{T^* \rightarrow \infty} \frac{L_T^*}{T^*} \leq \epsilon$.

Efficiency

M is efficient if $\liminf_{T^* \rightarrow \infty} \frac{T^*}{T} = 1$.

SAFEPREDICT GOAL: M should be *valid* for **any** P and be *efficient* when P performs well.

Background: Expert Advice and EWAF

- How to combine expert opinions P_1, \dots, P_N to perform almost as well as the best expert?

Exponentially weighted average forecasting (EWAF)

(Littlestone et al., 1989) (Vovk, 1990)

Intuition: Weight experts according to their past performances.

0. Initialize $(w_{P_1,1}, \dots, w_{P_N,1})$ and choose a learning rate $\eta > 0$.
 1. For each $t = 1, \dots, T$
 - 1.1. Follow P_i with probability $w_{P_i,t}$.
 - 1.2. Update the probability $w_{P_i,t+1} \propto w_{P_i,t} e^{-\eta l_{P_i,t}}$.
- **REGRET BOUND:**
$$L_T - \min_i L_{P_i,T} \leq \sqrt{T \log(N)/2}$$
where L_T and $L_{P_i,T}$ are the cumulative losses of EWAF and P_i .

Dummy and SafePredict

- We compare P with a **dummy predictor** (D) that refuses all the time.

$$l_{D,t} = \epsilon, \quad y_{D,t} = \emptyset.$$

- SafePredict is simply running EWAF over the ensemble $\{D, P\}$.
- EWAF regret bound implies $L_T^*/T^* - \epsilon = O(\sqrt{T}/T^*)$.



Therefore, for validity, we need a better bound and a more careful choice of η .

Theoretical Guarantees (Validity)

Theorem (Validity) ¹

Denoting the variance for the number of predictions with V^* and choosing $\eta = \Theta\left(1/\sqrt{V^*}\right)$, SafePredict is guaranteed to be valid for any P . Particularly,

$$\frac{L_T^*}{T^*} - \epsilon = O\left(\frac{\sqrt{V^*}}{T^*}\right) = O\left(\frac{1}{\sqrt{T^*}}\right),$$

where $V^* = \sum_{t=1}^T W_{P,t} W_{D,t}$.

¹ In practice, V^* can be estimated via so called “doubling trick”.

Theoretical Guarantees (Efficiency)

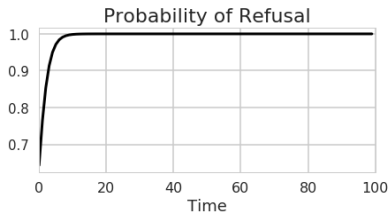
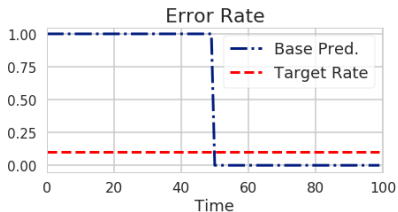
SafePredict is efficient as long as P has an error rate less than ϵ and η vanishes slower than $1/T$. Formally,

Theorem (Efficiency)

If $\limsup_{t \rightarrow \infty} L_{P,t}/t < \epsilon$ and $\eta T \rightarrow \infty$, then *SafePredict* is efficient.
Furthermore, the number of refusals are finite almost surely.

Weight Shifting

- Probability of making a prediction decreases exponentially fast if the base predictor has a higher error rate than ϵ . Therefore, it is hard to recover from long sequences of mistakes.
 - Probability of refusal only depends on the cumulative loss of P .
 - e.g. cold starts, concept changes.
- *Toy example:*

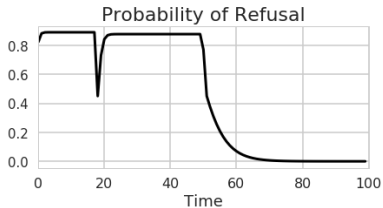
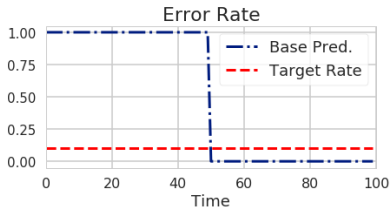


Weight Shifting

Weight-shifting: At each step, shift α portion of the D 's weight towards P , i.e.

$$w_{P,t} \leftarrow w_{P,t} + \alpha w_{D,t} = \alpha + (1 - \alpha)w_{P,t}.$$

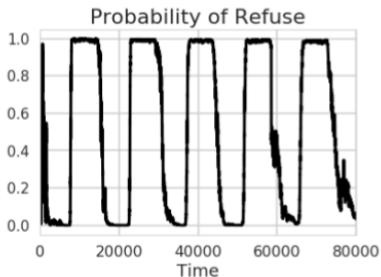
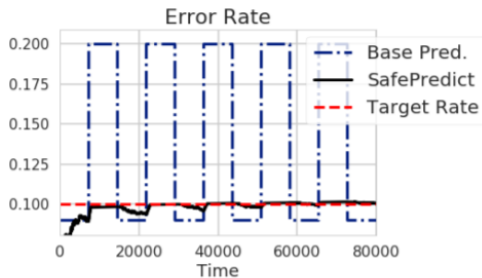
- Guarantees that $w_{P,t}$ is always greater than α .
- *Toy example:*



Weight Shifting

- Preserves the validity guarantee for $\alpha = O(1/T)$.
- Probability of refusal decreases exponentially fast if P performs better than D after t_0 .*

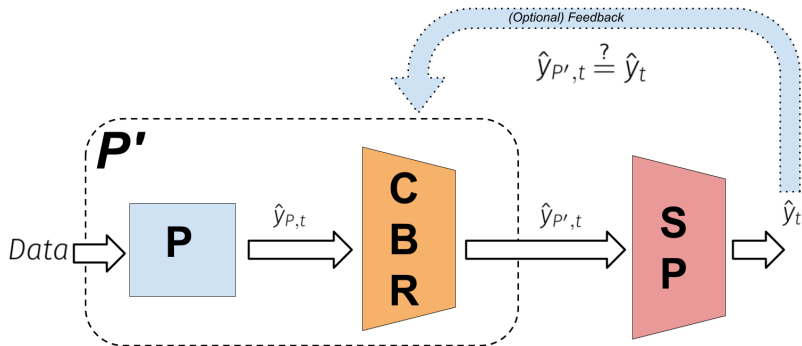
$$*w_{D,t} \leq e^{\eta \left(\sum_{\tau=t_0}^{t-1} l_{P,\tau} - \epsilon(t-t_0) \right)} / \alpha.$$



Hybrid Approach and Amnesic Adaptivity

- SafePredict uses only the loss values while deciding to refuse or predict. Therefore, it only infers when it is safe to predict.
 - **Robust validity** under any conditions.
- Conformity based refusal mechanisms (CBR) use the data itself and pick out the easy predictions *assuming* all the data points are coming from (roughly) the same distribution.
 - **Higher efficiency** when the data is i.i.d.
- **HYBRID APPROACH:** Employ SafePredict on top of other refusal mechanisms for the best of the both worlds.

Hybrid Approach and Amnesic Adaptivity



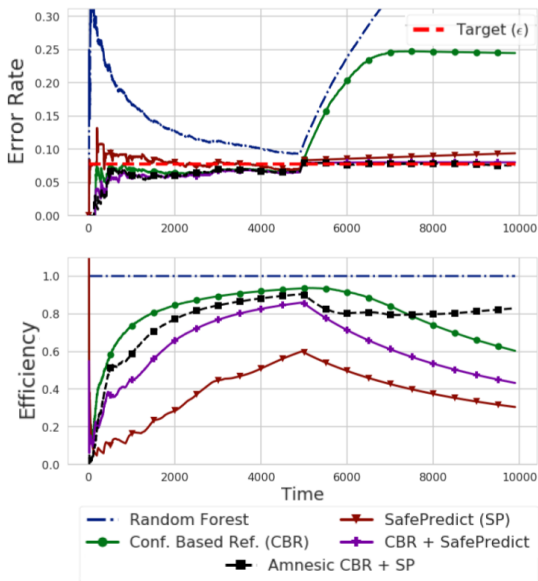
- If Confidence Based Refusal (CBR) mechanism predicts but SafePredict refuses, interpret this as violation of *i.i.d.* assumption.
 - **Amnesic adaptation:** if 50% of the last 100 predicted data points are refused by SafePredict, forget the history and reset the P' .

Numerical Experiments

Numerical Experiment (MNIST)

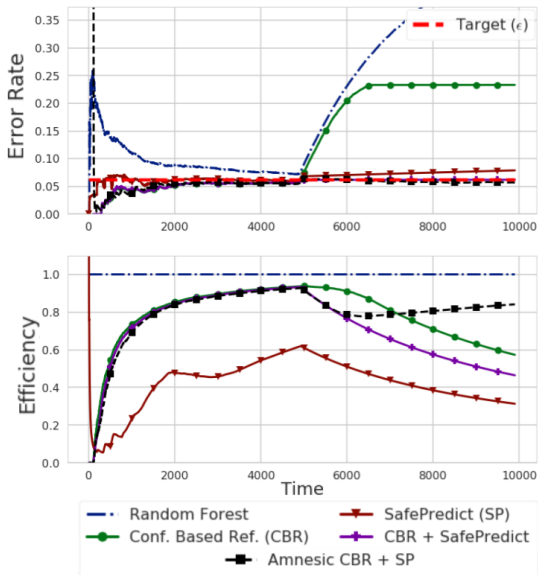
1 1 5 4 3
7 5 3 5 3
5 5 9 0 6
3 5 2 0 0

- $T = 10,000$.
- $\alpha = 10/T = 0.01$.
- P : Random forest retrained at every 100 data points.
- Change Point at $t = 5000$ (random permutation of labels).



Numerical Experiment (COD-RNA)

- Detection of non-coding RNAs (*Uzilov, 2006*)
- $T = 10,000$.
- $\alpha = 10/T = 0.01$.
- P : Random forest retrained at every 100 data points.
- Change Point at $t = 5000$ (random permutation of labels).



Conclusion

- We recast the exponentially weighted average forecasting algorithm to be used as a method to manage refusals.
- SafePredict works with any base prediction algorithm and asymptotically guarantees an upper bound on the error rate for non-refused predictions.
- The error guarantees do not depend on any assumption on the data or the base prediction algorithm.
- In changing environments, weight-shifting and amnesic adaptation heuristics boost efficiency while preserving the validity.
- Paper : <https://arxiv.org/abs/1708.06425>
I-Python Notebooks : <https://tinyurl.com/yagw3xzx>

Questions?

Back-up Slides

Conformity Based Refusals

1. Split the training set as **core training**, Z_1^n , and **calibration**, Z_{n+1}^{n+l} , sets where $n + l = m$.
2. Train the base classifier P on the **core training set**.
3. Choose the smallest threshold that gives an empirical error probability on the **calibration set** less than ϵ , i.e.

$$\tau^* = \inf \left\{ \tau : \frac{\sum_{i=m+1}^{m+l} 1_{\hat{y}_i \notin \{Y_i, \emptyset\}}}{\sum_{i=m+1}^{m+l} 1_{\hat{y}_i \neq \emptyset}} \leq \epsilon \right\}.$$

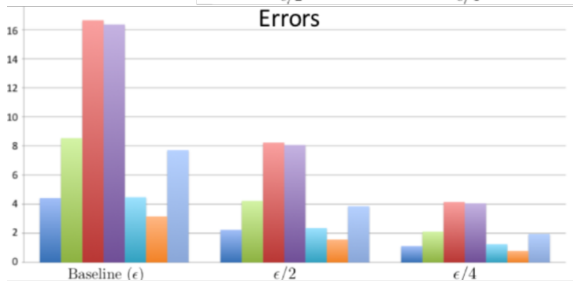
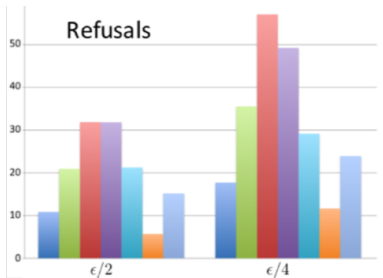
- This operation takes $O(l)$ computational time.

Then we have the following guarantee:

Theorem

We have $P_e \leq \epsilon + \frac{1}{1 - P_r} \sqrt{\frac{\log(l/\delta)}{2l}}$ with probability at least $1 - \delta$.

CBR: Experiments



SafePredict: Choosing the learning rate?

- Optimal learning rate: $\eta^* = K/\sqrt{V^*}$ for some constant $K > 0$.
- Use the “doubling trick” to estimate V^* .
- The validity guarantee is loosened by only a constant multiplicative factor of $\sqrt{2}/(\sqrt{2} - 1)$.

Weight-Shifting SafePredict with Doub. Trick

Base predictor: P ; Initial weight: $w_{P,1} \in (0, 1)$

Target error rate: $\epsilon \in (0, 1)$; Adaptivity Parameter: $\alpha \in [0, 1)$

1: Initialize $t = 1$

2: **for** each $k = 1, 2, \dots$ **do**

3: Reset $w_{P,t} = w_{P,1}$, $V_{sum} = 0$, and

$$\eta = \sqrt{-\log(w_{D,1} (1 - \alpha)^{T-1}) / (1 - \epsilon)^2 / 2^k}$$

4: **while** $V_{sum} \leq 2^k$ **do**

5: Predict with probability $w_{P,t}$, refuse otherwise,

$$\hat{y}_t = \begin{cases} \hat{y}_{P,t} & \text{with prob. } w_{P,t} \\ \emptyset & \text{otherwise} \end{cases}$$

6: Update the prediction probability:

$$w_{P,t+1} = \alpha + (1 - \alpha) \frac{w_{P,t} e^{-\eta l_{P,t}}}{w_{P,t} e^{-\eta l_{P,t}} + w_{D,t} e^{-\eta \epsilon}}$$

7: Compute $V_{sum} \leftarrow V_{sum} + w_{P,t+1} w_{D,t+1}$

8: Increment t by 1, i.e. $t \leftarrow t + 1$

Weight Shifting

Weight-shifting: At each step, shift α portion of the D 's weight towards P , i.e.

$$W_{P,t} \leftarrow W_{P,t} + \alpha W_{D,t}.$$

- Guarantees that $w_{P,t}$ is always greater than α .
- Preserves the validity guarantee for $\alpha = O(1/T)$.
- Probability of refusal decreases exponentially fast if P performs better than D after t_0 , i.e.

$$w_{D,t_1+1} \leq e^{\eta(L_{P,t_0,t_1} - \epsilon(t_1 - t_0))} / \alpha.$$

where $L_{P,t_0,t_1} = \sum_{t=t_0+1}^{t_1} l_{P,t}$ for any $t_0 < t_1$.