DriftFair: Anytime-Valid Temporal Fairness Monitoring and Shift Attribution for AI-Assisted Talent Promotion Decisions

Zimeng Wang[1], Zifan Chen[2,*], Zijian Shen[3]

[1]Brandeis University, Waltham, MA 02453, United States

[2]University of Pennsylvania, Philadelphia, PA 19104, United States

[3]Carnegie Mellon University, Pittsburgh, PA 15213, United States

[*]Corresponding author: Zifan Chen. jialeic@seas.upenn.edu

Abstract

Organisations increasingly deploy machine-learned scorers to shortlist employees for promotion. Such systems are typically audited for fairness once, at deployment, and then left unchecked, even though the population they judge keeps moving. We show that this practice is unsafe in a specific and measurable way: the drift that damages fairness and the drift that damages accuracy are largely disjoint, so accuracy-oriented monitors never fire, while input-oriented monitors fire on shifts that carry no fairness consequence at all. We present DriftFair, an operational monitor with four components. (i) A change detector for group-fairness functionals built as a weighted mixture of restarted betting e-processes; because the mixture is a non-negative supermartingale started at one, Ville's inequality gives a time-uniform false-alarm guarantee over an unbounded monitoring horizon, together with an explicit tolerance band that also absorbs the estimation error of the certified reference. (ii) An additive decomposition of any observed fairness drift into within-group covariate shift, latent (concept) shift and group-composition shift, obtained by per-group density-ratio reweighting, plus an exact Shapley attribution that names the responsible feature. (iii) A multi-cohort dashboard with online false-discovery-rate control via e-values. (iv) An attribution-driven maintenance policy. On a real promotion data set of 54,808 employee records, DriftFair is the only monitor among seven that combines a zero false-alarm rate on fairness-irrelevant drift with reliable detection of harmful drift; its shift attribution reaches 0.85 macro accuracy against 0.40 for a Kolmogorov-Smirnov input monitor, and its Shapley term recovers the perturbed feature with an attributed magnitude of 0.185 versus at most 0.021 for every other feature. Monitor-triggered per-group threshold recalibration cuts excess-disparity exposure by 48% relative to no maintenance while consuming 1,100 fresh labels, about one forty-ninth of the cheapest periodic schedule, and matches an oracle that knows the true change points. We further report a real chronological stream and a quantified data-sufficiency limit for small cohorts.

Keywords

Algorithmic fairness, concept drift, sequential change detection, anytime-valid inference, e-values, model monitoring, talent analytics.

1. Introduction

Promotion is one of the highest-stakes decisions an employer makes, and it is increasingly informed by learned scorers that rank internal candidates on performance ratings, training records, tenure and award history. Regulation has followed: New York City Local Law 144 requires annual bias audits of automated employment decision tools, and the European Union

AI Act classifies employment and worker-management systems as high risk and obliges providers to keep monitoring them after they go live. In practice, however, a fairness audit is still overwhelmingly a one-off artefact: a report produced on a certification sample at deployment time.

A one-off audit certifies a property of a joint distribution that does not stay still. Workforces reorganise, hiring channels change, training programmes are introduced, business units are merged, and the relationship between observed attributes and promotion outcomes is renegotiated. Any of these can move a certified demographic-parity or equal-opportunity gap outside the tolerance an organisation committed to, with no accompanying signal in the metrics that operations teams actually watch.

This paper starts from an empirical observation that makes the problem concrete rather than merely plausible. In our experiments, a latent reorganisation that leaves the observed feature distribution statistically unchanged moves the equal-opportunity gap from 0.051 to 0.196 while leaving classification error and the selection-rate gap essentially untouched; conversely, a covariate shift applied symmetrically to both groups moves the input distribution enough that a Kolmogorov-Smirnov monitor fires in 100% of runs, yet the fairness gap never leaves its tolerance band. Fairness drift and the drift that conventional monitors detect are therefore neither necessary nor sufficient for one another. A monitor that is going to be trusted by an employment review board has to be aimed at the fairness functional itself.

Aiming a monitor at a fairness functional raises three difficulties that existing tooling does not address together. First, fairness gaps are differences of proportions estimated on subgroup cohorts that are often small, so a naive dashboard that re-tests every review cycle accumulates false alarms; but the horizon is unknown, so a fixed-sample correction is also unavailable. Second, an alarm on its own is not actionable: the appropriate response to a shift in who applies is not the appropriate response to a shift in how outcomes are generated. Third, a real organisation monitors many cohorts at once, and cohort heterogeneity interacts with multiple testing in ways that can invalidate the whole dashboard.

Contributions. (1) We formulate temporal fairness monitoring as sequential testing of a tolerance null and construct a detector as a weighted mixture of restarted betting e-processes over review-cycle change points. The mixture is a non-negative supermartingale started at one, so Ville's inequality yields a time-uniform false-alarm guarantee that holds for an unbounded horizon; we also give a cycle-level Gaussian-mixture variant that is markedly more efficient when cohorts are large. (2) We show that the certified reference is itself an estimate and derive a tolerance band that absorbs its standard error, which is what separates a valid monitor from an invalid one when the certification sample is small. (3) We give an exact additive decomposition of fairness drift into within-group covariate, latent and composition components, with an exact Shapley attribution over features. (4) We show that per-cohort references combined with e-value online false-discovery-rate control are both necessary for a valid multi-cohort dashboard. (5) We evaluate on 54,808 real employee promotion records, 1,470 real HR records and a real two-year chronological decision stream, using a shift-injection protocol in which every label is real and only sampling propensities vary over time.

2. Related Work

A. Static fairness measurement and mitigation

Group-fairness criteria such as demographic parity and equality of opportunity [1], [2] and their impossibility relations [3], [4] are defined on a fixed distribution. Mitigation operates by preprocessing [5], constrained learning [6], [7] or post-hoc threshold adjustment [1], [8], and the design space is catalogued in survey form [9]. Open toolkits [10]-[12] institutionalise the static view: an audit is a computation performed on a data set at a point in time, and reporting

conventions such as model cards [13] document that computation rather than its persistence. Domain studies of algorithmic hiring [14], [15], the legal doctrine of disparate impact [16] and documented deployment failures [17] motivate the setting but likewise treat auditing as episodic.

B. Drift detection and monitoring

Concept-drift detection is mature [18], [19]. ADWIN maintains an adaptive window and cuts it when two sub-windows differ beyond a Hoeffding bound [20]; DDM tracks the online error rate [21]; the Page-Hinkley test accumulates a CUSUM statistic [22]. Multivariate two-sample tests detect input shift [23], [24], and black-box shift estimation recovers label shift [25]; benchmarks rebuilt from successive survey years show how large genuine temporal shift can be [26]. All of these monitor accuracy or the input distribution; our experiments quantify how weakly either target relates to fairness drift. Production-ML checklists [27], [28] recommend continuous monitoring without specifying a fairness-valid procedure.

Two recent studies make the cost of the static view concrete in deployed systems. Rhee et al. [29] show that web-search agents fine-tuned on static trajectory snapshots degrade once real web pages change structurally, and demonstrate that a self-play loop which keeps contrasting the agent's own successes against its own failures recovers a substantial part of that loss; their practice of reporting unperturbed and perturbed performance separately, rather than a single headline number, is a template we adopt. Fan et al. [30] press the point further, arguing that in financial fraud detection the drift is not stochastic but strategic, and showing that the standard industrial answer of periodic retraining buys adaptation to new patterns at the price of forgetting older ones. Both works respond to change by changing the model; DriftFair is complementary, leaving the deployed decision rule fixed and instead telling an operator, with a false-alarm guarantee, when and why a specific certified property has stopped holding.

C. Fairness over time

Simulation studies show that static fairness guarantees decay under feedback and population change [31], [32]. Fairness-aware stream learners adapt a classifier online [33]-[35] but do not provide a monitoring guarantee, and they change the model rather than inform an operator. FairCanary [36] is the closest system: it computes a quantile-based bias metric continuously with feature-level explanations. It does not, however, control the false-alarm rate over an unbounded horizon, does not separate covariate from latent causes, and does not address multi-cohort multiplicity.

D. Anytime-valid inference and e-values

Testing by betting yields confidence sequences for bounded means [37] and safe anytime-valid inference [38]. E-values support calibration, combination and false-discovery-rate control [39]-[41], and e-detectors reduce sequential change detection to sequential testing by summing e-processes started at consecutive times [42]. Sequential fairness auditing has been studied for a fixed audited quantity [43]-[45]. Two adjacent lines of work show what it takes to carry such guarantees into a moving world. Chen et al. [46] operationalise distribution-free coverage for non-stationary financial series by normalising the nonconformity score with a local volatility estimate and recalibrating on a rolling, time-decayed window, and they are commendably explicit that what survives deployment is approximate rather than exact validity. Liang et al. [47] address the complementary problem of attribution, decomposing simultaneous supply and demand movements in a non-stationary system into separately identified components while keeping inference valid when the identifying signal is weak. Our contribution is not a new e-process but the construction of an operational monitor: a tolerance null that encodes an organisational commitment and the uncertainty of its certification, a decomposition that turns an alarm into a diagnosis, a dashboard layer with online error control, and a maintenance policy driven by the diagnosis.

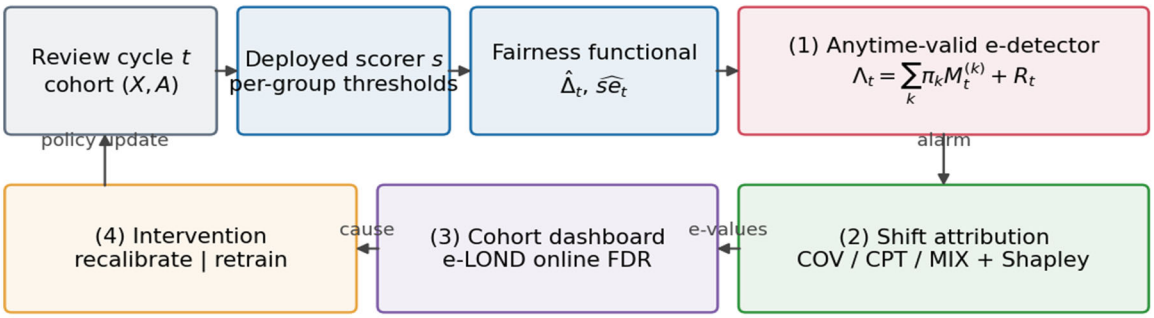

Figure 1. The DriftFair monitoring loop. A frozen scorer produces decisions for each review cohort; the fairness functional and its standard error feed an anytime-valid e-detector; an alarm is decomposed into covariate, latent and composition components and attributed to features; cohort-level e-values are combined under online false-discovery-rate control; the diagnosis selects the intervention.

3. Problem Setting

An organisation reviews candidates in discrete cycles t = 1, 2, ... Cycle t draws a cohort of n_t candidates with features X, protected attribute A in {0,1} and a later-observed outcome Y in {0,1} (promoted and retained). A frozen scorer s recommends the top of the cohort subject to a head-count budget, giving decisions Yhat = 1{s(X) >= tau_A}. We monitor two functionals:

$$Delta\_t\text{^}DP = P\_t(Yhat=1 \mid A=1) - P\_t(Yhat=1 \mid A=0), \quad (1)$$

$$Delta\_t\text{^}EO = P\_t(Yhat=1 \mid A=1, Y=1) - P\_t(Yhat=1 \mid A=0, Y=1). \quad (2)$$

The selection-rate gap (1) needs no labels and is available immediately; the equal-opportunity gap (2) needs outcomes and is therefore delayed. This distinction matters: because the scorer is frozen, (1) depends only on the within-group feature distributions P_t(X | A), so a shift confined to P(Y | X) is invisible in (1) and visible only in (2). Our experiments confirm this exactly.

At deployment the organisation certifies a reference value Delta_ref on a certification sample and commits to a tolerance epsilon. The monitoring problem is the sequential test of

$$H\_0 : \ |Delta\_t - Delta\_ref| <= epsilon \quad for\ all\ t, \quad (3)$$

with a false-alarm probability controlled uniformly over an unbounded horizon, plus the diagnostic problem of explaining any rejection.

4. The DriftFair Framework

A. An anytime-valid fairness e-detector

Fix a candidate change point k. For observations W_i in [0,1] with conditional mean mu_i, the betting capital process M_t^(k) = prod_{i=k..t} (1 + lambda_i (W_i - mu_0)) with predictable, suitably truncated lambda_i >= 0 is a non-negative supermartingale whenever mu_i <= mu_0 [37]. Because the change point is unknown we combine the restarts with weights pi_k that sum to one:

$$Lambda_t = sum_{k<=t} pi_k M_t^(k) + sum_{k>t} pi_k. \quad (4)$$

Proposition 1. If every M^(k) is a non-negative supermartingale under H_0 and sum_k pi_k = 1, then Lambda in (4) is a non-negative supermartingale with Lambda_0 = 1, and consequently

$$P_{H_0} (there exists t : Lambda_t >= 1/alpha) <= alpha. \quad (5)$$

Proof sketch. Setting M_t^(k) = 1 for t < k gives Lambda_0 = sum_k pi_k = 1. Conditional expectations of the summands are bounded by their previous values, so Lambda is a supermartingale; Ville's inequality gives (5). The statement is uniform in t and needs no horizon.

Two properties make (4) practical. First, restricting change points to review-cycle boundaries and taking pi_k proportional to 1/(k(1+log k)^2) keeps the weights large while remaining summable, so no horizon has to be declared. Second, with a shared predictable lambda_t the whole mixture obeys the O(1) recursion S_t = f_t (S_{t-1} + pi_t), Lambda_t = S_t + R_t, where f_t is the current betting factor and R_t is the unspent weight.

We instantiate (4) in two ways. DriftFair-B pairs one member of each group inside a cycle, so that the paired difference Z in {-1,0,1} has mean Delta_t, and bets on W = (Z+1)/2 with a GRAPA-style predictable plug-in; it is finite-sample valid and assumption-free. DriftFair-G aggregates each cycle into one standardised residual u_t = (Deltahat_t - mu_0)/sehat_t, mixes exp(lambda u_t - lambda^2/2) over a grid of lambda, and plugs the result into (4). DriftFair-G is only asymptotically valid, but because a cohort aggregates hundreds of decisions it is far more efficient; we therefore recommend it whenever both subgroup cohorts exceed roughly one hundred members and DriftFair-B otherwise. Both sides of the two-sided test use alpha/2.

B. The reference is an estimate

Delta_ref is measured on a finite certification sample and carries a standard error se_ref. Testing against the nominal band [Delta_ref - epsilon, Delta_ref + epsilon] therefore tests a null that may be false at deployment, and no amount of sequential validity repairs this. We widen the band to

$$epsilon_eff = epsilon + z se_ref, \quad (6)$$

with z = 2 by default, so that the null holds with high probability over the certification draw. The same failure mode has been identified in conformal prediction for non-stationary series: Chen et al. [46] show that a calibration quantile computed once silently loses coverage unless the nonconformity score is normalised by a moving scale estimate and the quantile is refreshed on a rolling window. Equation (6) is the corresponding correction for a fairness functional, applied to the reference rather than to the score, and Section VI shows that on a certification sample of 16,442 records the inflation costs detection power, whereas on 441 records it is the difference between a usable monitor and one that alarms on every drift-free stream.

C. Decomposing a fairness alarm

Let R be the certification sample and W a detection window. Write Delta(W) for the gap on W and Delta(W -> R) for the gap on W after reweighting, inside each protected group, so that the observed feature distribution matches R. Then identically

$$Delta(W) - Delta(R) = COV + CPT, \quad (7)$$

$$COV = Delta(W) - Delta(W -> R), \quad CPT = Delta(W -> R) - Delta(R). \quad (8)$$

COV is the part of the drift explained by movement of the observed covariates within groups; CPT is the residual, attributable to changes in the outcome mechanism or to latent variables outside X. A third quantity MIX = P_W(A=1) - P_R(A=1) records composition change; because (1) and (2) condition on A, composition change alone does not move them, and reporting MIX separately is what prevents a harmless reorganisation from being escalated. Weights come from per-group domain classifiers, clipped to a bounded range. A shift is declared negligible when |Delta(W) - Delta(R)| falls below twice the standard error of that difference, which ties the decision to sampling noise rather than to a hand-set constant.

Because COV is defined through an alignment operation on a set of features, its attribution over features is a cooperative game: for a subset S let v(S) be the reduction in the gap obtained by aligning only the features in S. With nine features the exact Shapley values of v are computable over all 512 subsets, which names the feature responsible for the drift rather than merely flagging that some input moved.

Decomposing an observed movement of a non-stationary system into separately identified components is a general problem, and Liang et al. [47] solve a demanding version of it in macro-econometrics by constructing external instruments that isolate one causal channel at a time. Our setting admits a sharper device: because the decision function is frozen and known, aligning the observed covariates is a reweighting rather than an estimation problem, so (7) holds as an identity instead of as the output of a fitted system.

D. Cohort dashboards

An organisation monitors K cohorts, so the final value Lambda_T of each detector is used as an e-value (its expectation is at most one under H_0) and the cohorts are processed by e-LOND [41], which controls the false-discovery rate under arbitrary dependence. Section VI shows that the choice of reference matters more than the correction: with one organisation-wide reference, cohort heterogeneity makes the nulls false and no correction restores validity.

E. Attribution-driven maintenance

On an alarm DriftFair computes (7)-(8) and chooses an action. If COV dominates, the decision rule is misaligned with a moved population and per-group thresholds are re-fitted on a small freshly labelled batch: tau_1 is swept over a grid and tau_0 is set so that the total head count meets the budget exactly, which turns a two-dimensional search into a one-dimensional one and always respects the constraint. If CPT dominates, the outcome mechanism has moved and the scorer is retrained on recent cycles. The certified reference is never reset by an intervention, so that disparity cannot ratchet upwards by repeatedly redefining what is normal.

Which of the two actions is chosen matters more than it may appear. Retraining is the default industrial response to drift, and Fan et al. [30] quantify its cost in a fraud-detection setting, where refitting on recent data overwrites competence on older attack patterns unless the replay buffer is curated deliberately. Section VI-C reports the fairness analogue of that trade, in which retraining improves accuracy and simultaneously widens the very gap the operator is trying to close.

Table I. Real data sets and deployed systems.

Data set	Records	Feat.	Protected	P(A=1)	Base rate	AUC	n / cycle
PROMO (WNS employee promotion)	54,808	9	gender	0.298	0.085	0.833	1,200
IBM-HR (IBM HR analytics)	1,470	9	gender	0.400	0.361	0.702	200
COMPAS (ProPublica, 2013-2014)	5,278	7	race	0.600	0.451	0.682	~98

Records are after the standard filters. Base rate is P(Y = 1). AUC is measured on the deployment pool with the frozen scorer. COMPAS cycles are calendar-ordered.

5. Experimental Setup

A. Data

We use three real data sets (Table I). PROMO is the WNS Analytics employee-promotion data set, 54,808 employee records with nine features, gender as the protected attribute and a real promotion outcome; it also contains 34 organisational regions whose promotion rates range from 1.9% to 14.4%, which we use as a latent variable. IBM-HR is IBM's published HR analytics extract, 1,470 records, used as a small-cohort stress test with the target defined as promotion in the current year. COMPAS is the ProPublica release [17], which after the standard filters yields 5,278 records carrying real screening dates from April 2013 to December 2014 and therefore a genuine chronological decision stream.

B. Deployment streams and shift injection

A gradient-boosted scorer is fitted on a certification split and frozen; the threshold is the 85th score percentile, i.e. a 15% promotion budget. Review cycles are drawn from the disjoint deployment pool. Every label in every stream is a real label: we never relabel. Non-stationarity is induced only through time-varying sampling propensities, which gives three regimes with known ground truth. Covariate shift tilts sampling by an observed feature, which leaves P(Y | X, A) unchanged by construction. Latent (concept) shift tilts sampling by the organisational region, a variable excluded from X, and does so within strata of X so that the observed marginal is held fixed while P(Y | X) genuinely moves; measured on the promotion stream, mean avg_training_score moves from 63.23 to 63.15 while the equal-opportunity gap moves from 0.072 to 0.184. Composition shift changes P(A). A benign covariate shift applies the same tilt to both groups, so the input distribution moves while the fairness gap stays inside tolerance. Shifts ramp over six cycles from cycle 35 of 100.

C. Baselines, protocol and metrics

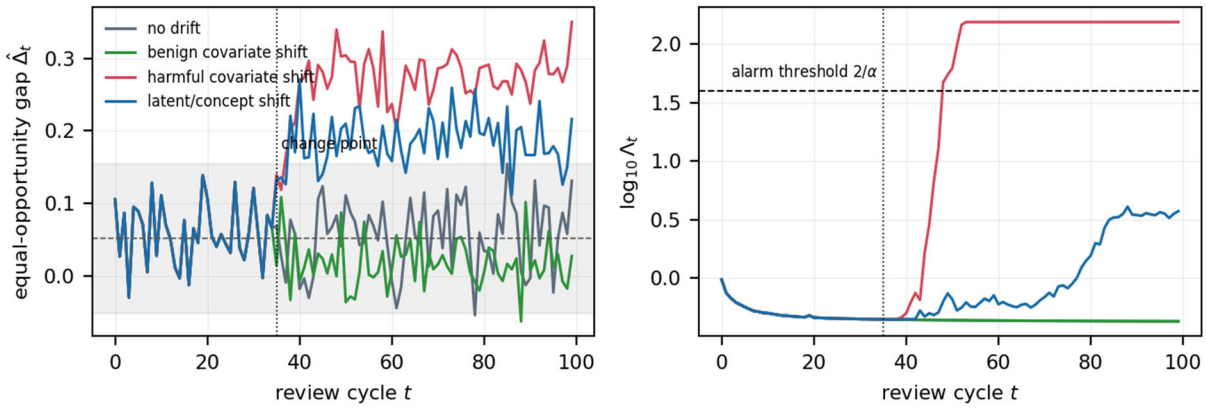

Figure 2. PROMO, equal-opportunity channel. Left: mean realised gap per cycle; the shaded band is the tolerance band of (6) and the dotted line marks the change point. Right: the monitoring statistic log10 Lambda_t against the alarm threshold 2/alpha. The benign covariate shift moves the gap without leaving the band and the statistic stays flat; the harmful covariate and latent shifts drive it upwards.

We compare against ADWIN [20] and Page-Hinkley [22] applied to the monitored fairness series, DDM [21] and ADWIN applied to the error stream (fairness-blind monitors), a per-cycle two-proportion z-test with Bonferroni correction over the horizon, a per-feature Kolmogorov-

Smirnov input monitor in the spirit of [23], and a non-restarting betting audit corresponding to sequential fairness auditing without change-point mixing [43]. Every baseline sensitivity knob is calibrated on drift-free streams so that its false-alarm rate over the horizon does not exceed alpha = 0.05; DriftFair uses its theoretical threshold and is never tuned. We report the detection rate and mean delay on harmful drift, the alarm rate on drift-free and fairness-irrelevant streams, and for maintenance the excess-disparity exposure sum_t n_t max(0, |Delta_t - Delta_ref| - epsilon), which counts candidate-cycles judged under an out-of-tolerance rule. Throughout epsilon = 0.05 and alpha = 0.05. Following the reporting practice of Rhee et al. [29], we give per-condition alarm rates and delays for every run and refrain from attaching significance claims to differences that our number of repetitions cannot support.

Table II. PROMO: detection rate and mean delay in review cycles (10 runs, epsilon = 0.05, alpha = 0.05). Baseline knobs are calibrated so that the drift-free alarm rate does not exceed alpha.

Method	no drift	benign cov.	composition	harmful cov. (DP)	harmful cov. (EO)	latent (DP)	latent (EO)
DriftFair-G (ours)	0.00	0.00 / 0.00	0.00	1.00 (11.0)	1.00 (14.0)	0.00	0.20 (46.5)
DriftFair-B (ours)	0.00	0.00 / 0.00	0.00	0.80 (35.5)	1.00 (13.1)	0.00	0.00
Betting audit, no restart	0.00	0.00 / 0.00	0.00	0.00	1.00 (12.2)	0.00	0.00
z-test + Bonferroni	0.00 / 0.10	0.00 / 0.10	0.00 / 0.10	0.90 (21.0)	0.80 (23.2)	0.00	0.10 (25.0)
ADWIN on gap series	0.00	0.00 / 0.00	0.00	0.00	0.70 (40.6)	0.00	0.00
Page-Hinkley on gap series	0.00 / 0.10	1.00 / 0.20	0.00 / 0.10	1.00 (8.3)	1.00 (15.8)	0.00	0.70 (21.7)
KS input monitor	0.00	1.00 / 1.00	0.00	1.00 (2.6)	1.00 (2.6)	0.10	0.10
DDM on error stream	0.00	0.80 / 0.80	0.00	0.20 (32.0)	0.20 (32.0)	0.00	0.00
ADWIN on error stream	0.00	0.00 / 0.00	0.00	0.00	0.00	0.00	0.00

Cells are alarm rates; mean detection delay in cycles is in parentheses where the method detects. Where two numbers appear they are the selection-rate (DP) and equal-opportunity (EO) channels. The first three columns are streams that must not raise an alarm: no drift at all, a covariate shift applied to both groups that leaves the gap inside tolerance, and a change of P(A) alone. Lower is better in those columns, higher is better in the last four.

6. Results

A. Fairness drift and conventional drift are different things

Table II is the central result. Under a harmful covariate shift DriftFair-G detects in 100% of runs on both channels, with mean delays of 11.0 and 14.0 cycles against 21.0 and 23.2 for the Bonferroni-corrected z-test; ADWIN on the gap series never fires on the selection-rate channel and fires in 70% of runs on the equal-opportunity channel with a delay of 40.6 cycles. Under latent shift the selection-rate channel is silent for every method, which is the behaviour predicted in Section III: with a frozen scorer, the label-free gap cannot see a change confined to P(Y | X). Only the outcome-conditioned channel responds. This is a practical warning, because label-free monitoring is exactly what most deployments can afford.

The right-hand panel of Fig. 3 is the complement. Under a benign covariate shift the Kolmogorov-Smirnov input monitor fires in 100% of runs and DDM in 80%, while the realised gap never leaves its tolerance band; Page-Hinkley, calibrated to 5% on drift-free streams, also fires in 100% of runs on the selection-rate channel. DriftFair fires in none of them. Calibrating a change detector on stationary streams therefore says nothing about its behaviour under shifts that are real but harmless, and this is where alarm fatigue is manufactured. Taking the

difference between detection rate on harmful drift and alarm rate on nuisance drift, DriftFair is the only method with a positive margin on every channel.

Table III repeats the study on IBM-HR, where subgroup cohorts contain tens rather than hundreds of members. Here the reference inflation of (6) is decisive: the certified equal-opportunity gap has a standard error of 0.072 on 441 records, so the nominal band is narrower than the uncertainty of its own centre. With inflation the drift-free alarm rate is 0.0 and covariate shift is still detected in 100% of runs at a delay of 26.2 cycles; without it the monitor alarms on drift-free streams. Latent shift is not detectable at this cohort size, which is a data-sufficiency statement rather than an algorithmic failure.

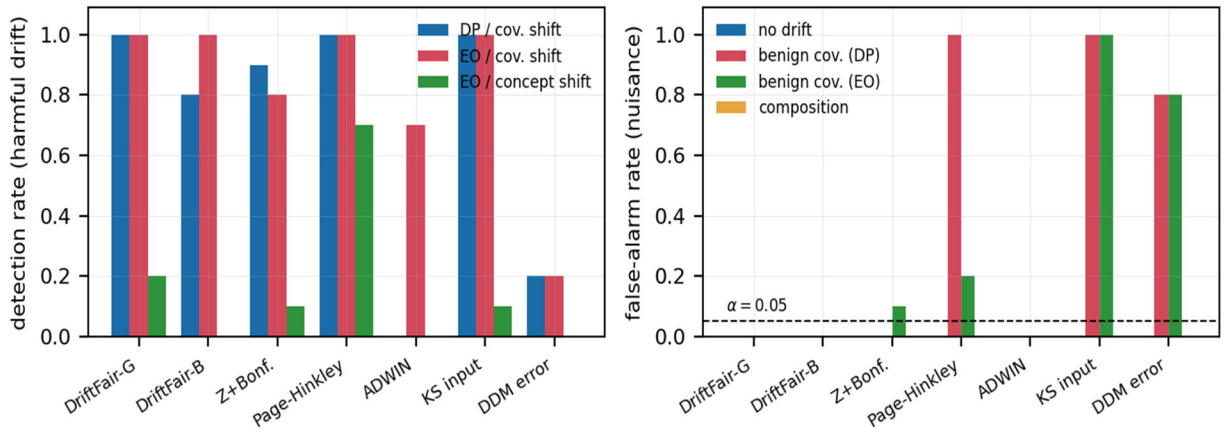

Figure 3. PROMO. Left: detection rate on harmful drift. Right: alarm rate on streams that must not raise an alarm. Only DriftFair combines high power with zero nuisance alarms; monitors calibrated on stationary streams still fire on real but harmless shifts.

Table III. IBM-HR small-cohort regime (10 runs). The certified equal-opportunity gap has standard error 0.072 on 441 records.

Method	no drift (DP/EO)	benign cov. (EO)	harmful cov. (DP)	harmful cov. (EO)
DriftFair-G with band (6)	0.00 / 0.00	1.00	0.20 (53.5)	1.00 (26.2)
DriftFair-B with band (6)	0.00 / 0.00	1.00	0.00	1.00 (26.9)
z-test + Bonferroni	0.00 / 0.00	0.70	0.10 (16.0)	0.40 (46.5)
Page-Hinkley on gap series	0.00 / 0.00	1.00	1.00 (26.1)	1.00 (21.8)
KS input monitor	0.00 / 0.00	1.00	1.00 (3.6)	1.00 (3.6)
DDM on error stream	0.00 / 0.00	0.00	0.00	0.00

Without the inflation of (6) the equal-opportunity monitor alarms on drift-free streams, because the nominal tolerance band is narrower than the standard error of its own centre. The benign-shift column is elevated for every fairness monitor at this cohort size, which is the honest cost of monitoring tens rather than hundreds of decisions per cycle.

B. Attribution recovers the cause and the feature

Table IV reports the decomposition. Injected covariate shift yields COV = 0.222 against CPT = 0.002; injected latent shift yields COV = 0.023 against CPT = 0.121; composition shift leaves both near zero and shows up in MIX = 0.253. The resulting classifier is correct in 12/12, 11/12 and 12/12 runs on these three regimes and in 12/12 on drift-free streams, for a macro accuracy

of 0.850. A Kolmogorov-Smirnov heuristic reaches 0.400 because it cannot see a latent shift that preserves the observed marginal and cannot represent composition change; an error-rate heuristic reaches 0.200 because it labels everything a concept shift. The genuinely mixed regime is recovered as mixed in only 4/12 runs and as concept-dominant in 6/12, which is honest about a case in which the two components are of comparable size.

The Shapley panel of Fig. 4 is a sharper test, because the ground truth is a single perturbed feature. Under covariate shift the attribution assigns 0.185 to avg_training_score, the feature actually tilted, and at most 0.021 to any other feature. Under latent shift no feature dominates and the attributed value of avg_training_score is slightly negative, which is the correct answer: the cause is not in X.

Table IV. PROMO: shift attribution (12 runs per regime). Decomposition values are means; classification counts are out of 12.

Injected regime	COV	CPT	MIX	DriftFair correct	KS heuristic	error heuristic
none	+0.015	-0.002	+0.001	12 / 12	12 / 12	0 / 12
within-group covariate	+0.222	+0.002	+0.001	12 / 12	12 / 12	0 / 12
latent (concept)	+0.023	+0.121	-0.000	11 / 12	0 / 12	12 / 12
composition	+0.010	+0.006	+0.253	12 / 12	0 / 12	0 / 12
mixed	+0.037	+0.075	-0.000	4 / 12	0 / 12	0 / 12
macro accuracy				0.850	0.400	0.200

COV, CPT and MIX are the components of (7)-(8) on the equal-opportunity gap. The Kolmogorov-Smirnov heuristic cannot express a latent shift that preserves the observed marginal; the error heuristic labels every regime a concept shift. The mixed regime perturbs both components at comparable magnitude and is intrinsically ambiguous.

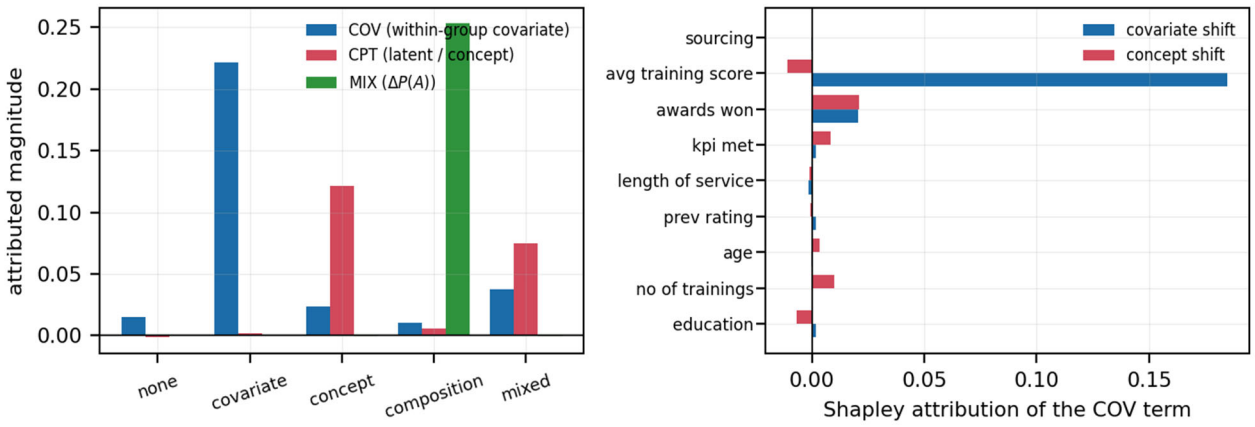

Figure 4. PROMO. Left: the decomposition (7)-(8) recovers the injected cause in every regime. Right: exact Shapley attribution of the COV term over features. Under covariate shift the attribution isolates avg_training_score, the feature actually perturbed; under latent shift no feature carries the drift, which is the correct diagnosis.

C. Maintenance: the right action matters more than the right moment

Table V evaluates maintenance over a horizon with a covariate shift at cycle 28 and a latent shift at cycle 64. Three findings stand out. First, retraining improves accuracy and damages fairness: periodic retraining every ten cycles raises mean AUC from 0.851 to 0.863 while raising mean absolute deviation from 0.171 to 0.217 and consuming 54,000 labels. Refitting on drifted data reproduces the drifted relationship; nothing in the objective protects the gap, which mirrors in a fairness metric the accuracy-side trade Fan et al. [30] report when a fraud detector is retrained on recent data. Second, the accuracy-triggered policy never triggers at all, because the injected drift does not hurt accuracy; its row is identical to no maintenance. Third, DriftFair-triggered per-group recalibration attains a mean deviation of 0.107 and an exposure of 6,482, a 48% reduction against no maintenance, using 1.8 interventions and 1,100 labels: one forty-ninth of the cheapest periodic recalibration schedule at a lower exposure, one hundred and thirteenth of the retraining schedule, and statistically indistinguishable from an oracle told the true change points (0.106, 6,383). The attribution-aware variant lands between the two because it correctly routes the second, latent shift to retraining, which restores accuracy (0.853) at a higher label cost. Fig. 5 shows the trade-off and the trajectory.

Table V. PROMO: maintenance policies over 100 cycles with a covariate shift at cycle 28 and a latent shift at cycle 64 (6 runs, cohorts of 1,000).

Policy	mean |dev.|	exposure	viol. rate	AUC	retrains	recalibs.	labels
no maintenance	0.171	12,512	0.847	0.851	0.0	0.0	0
retrain every 10 cycles	0.217	17,007	0.883	0.863	9.0	0.0	54,000
retrain every 20 cycles	0.197	15,037	0.870	0.860	4.0	0.0	24,000
recalibrate every 10 cycles	0.154	10,862	0.807	0.851	0.0	9.0	5,400
recalibrate every 20 cycles	0.140	9,544	0.773	0.851	0.0	4.0	2,400
accuracy-triggered (ADWIN)	0.171	12,512	0.847	0.851	0.0	0.0	0
DriftFair -> retrain	0.206	15,983	0.865	0.863	20.7	0.0	124,000
DriftFair -> recalibrate	0.107	6,482	0.697	0.851	0.0	1.8	1,100
DriftFair -> attribution-aware	0.120	7,695	0.718	0.853	2.5	1.7	16,000
oracle change-point timing	0.106	6,383	0.690	0.851	0.0	2.0	1,200

Mean |dev.| is the mean absolute deviation of the equal-opportunity gap from its certified value; exposure is the excess-disparity exposure defined in Section V-C; viol. rate is the fraction of cycles outside tolerance; labels counts freshly labelled records consumed. Standard deviations over runs are at most 0.034 for mean |dev.|. The accuracy-triggered policy never fires, so its row coincides with no maintenance.

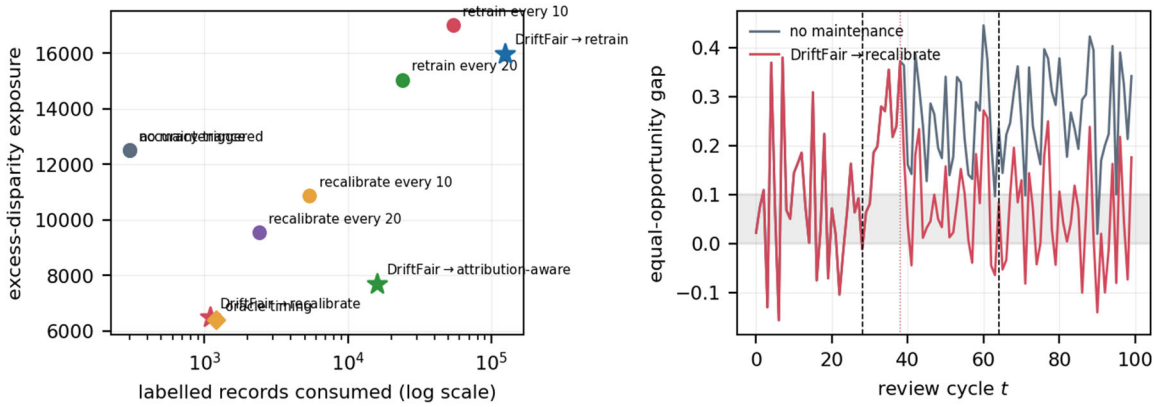

Figure 5. PROMO maintenance. Left: excess-disparity exposure against labelling cost; stars are DriftFair-triggered policies and the diamond is the oracle. Right: the equal-opportunity trajectory without maintenance and under DriftFair-triggered recalibration; dashed lines are the true change points and dotted lines the interventions actually taken.

D. Multi-cohort dashboards and the choice of reference

Table VI monitors fourteen real department-by-education cohorts with drift injected into four. Auditing every cohort against one organisation-wide reference gives a false-discovery rate of 0.576 uncorrected, and 0.643 under e-LOND: the correction reduces the number of rejections but concentrates them on the cohorts whose baseline gap differs most from the global value, so the errors are systematic rather than random and multiplicity control cannot remove them. Estimating a reference per cohort and then applying e-LOND halves the false-discovery rate to 0.299 and raises power from 0.30 to 0.56. The lesson for practice is that a dashboard must model cohort heterogeneity before it worries about multiple testing.

E. A real chronological stream, and how much data is enough

On COMPAS the reference is measured over an eighteen-cycle burn-in of the live system (Delta_ref^DP = 0.197, Delta_ref^EO = 0.184) and monitoring runs over the remaining twenty-two cycles of about 98 decisions each. In the arm where nothing changes, DriftFair raises no alarm on either channel over the whole stream, while the Kolmogorov-Smirnov monitor fires at cycle 36; the large baseline disparity is a static fairness problem and correctly outside the remit of a drift monitor. In the arm where the deployed scorer is replaced by one that no longer uses priors_count, the equal-opportunity gap moves from 0.184 to 0.119 and the decomposition returns COV = -0.008 against CPT = -0.056, i.e. it reports correctly that no data shift occurred and the change originates in the system itself. The change is nevertheless not certified at epsilon = 0.05: a deviation of 0.065 on cohorts of this size does not accumulate enough evidence in twenty-two cycles.

Fig. 6 turns this into a usable rule. The left panel is the detection rate of DriftFair-G against the realised deviation on the promotion stream. With cohorts of 1,200 and the inflated band, detection is reliable once the realised deviation reaches roughly four times epsilon and is essentially certain beyond that; without inflation the same curve shifts left by about one epsilon, with no measurable cost in drift-free alarms on this certification split. Because the evidence rate per cycle is approximately (excess deviation)^2 / (2 var(Deltahat_t)), the number of cycles needed scales as log(1/(pi_k alpha)) divided by that rate, which lets an operator convert a cohort size and a tolerance into a monitoring horizon before deployment. The right panel shows the evidence accumulating in the model-swap arm and staying flat in the unchanged arm on the real stream.

Table VI. PROMO: dashboard over 14 real department-by-education cohorts, drift injected into 4 (25 replications, target FDR 0.10).

Dashboard design	FDR	power	rejections
organisation-wide reference, uncorrected	0.576	0.430	3.92
organisation-wide reference, e-LOND	0.643	0.300	3.00
per-cohort reference, e-LOND (ours)	0.299	0.560	3.28

With a single organisation-wide reference the nulls are false for heterogeneous cohorts, so the errors are systematic and multiplicity control cannot remove them; it reduces rejections while concentrating them on the most heterogeneous cohorts.

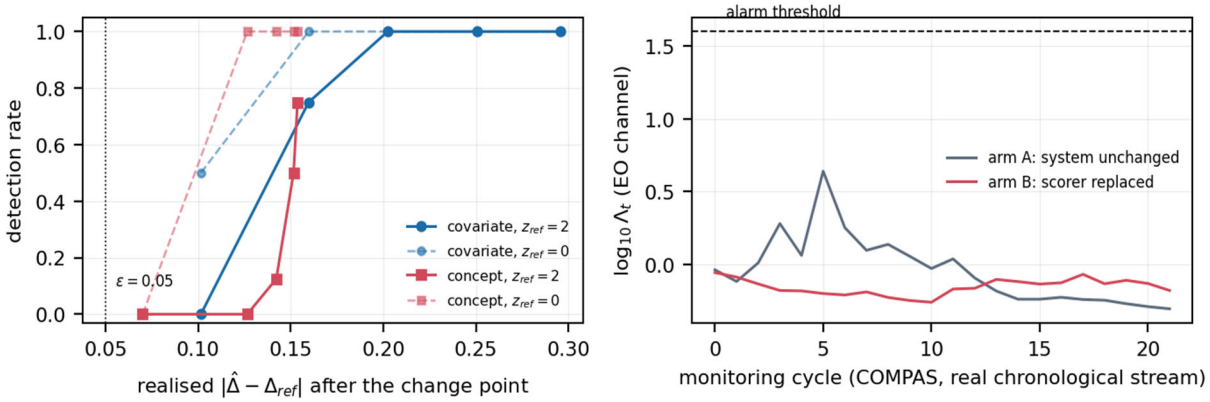

Figure 6. Left: detection rate of DriftFair-G against the realised deviation on PROMO, with and without the reference inflation of (6); the vertical line is epsilon. Right: the monitoring statistic on the real COMPAS chronological stream. Evidence accumulates after the deployed scorer is replaced and stays flat when the system is unchanged.

7. Discussion and Limitations

DriftFair monitors a functional of a frozen decision rule; it is not a fairness intervention and cannot certify that the certified reference was itself acceptable. The tolerance null encodes an organisational commitment, which is a policy choice that our method makes explicit rather than resolves. The gap between DriftFair-B and DriftFair-G is a genuine trade-off between finite-sample validity and efficiency, and the Gaussian variant should not be used when subgroup cohorts fall to a few dozen members; the honest description of what it delivers, as Chen et al. [46] put it for conformal intervals under drift, is approximate validity under moderate non-stationarity rather than an exact guarantee. The reference inflation of (6) is conservative by design: on a large certification sample it costs detection power, as Fig. 6 quantifies, and an operator who can afford a large certification sample should reduce z rather than accept the loss. Latent shift is detected only through the outcome-conditioned channel and therefore inherits the outcome-reporting delay, which in promotion settings can be a full review cycle or more. Our shift-injection protocol keeps every label real but still constructs the arrival process, so the promotion results should be read as a controlled study whose ground truth is known, complemented by the uncontrolled real stream of Section VI-E. Finally, protected attributes are assumed observed and binary; intersectional monitoring multiplies the number of cohorts and makes the dashboard layer of Section IV-D indispensable rather than optional. A further boundary is that we treat drift as exogenous. Once candidates or line managers can observe the deployed rule and adapt to it the drift becomes strategic, and Fan et al. [30] show that in that regime a defender has to anticipate the adaptation rather than react to it; extending our

tolerance null to such a performative setting is open. The obligation being served here is finally regulatory as much as ethical, since employment and worker-management systems are classified as high risk under the EU AI Act [48], which requires post-market monitoring rather than a single pre-deployment audit.

8. Conclusion

Fairness certified once is not fairness maintained. We showed that the drift which damages a promotion system's fairness is largely orthogonal to the drift that accuracy and input monitors are built to catch, and we gave a monitor aimed at the fairness functional itself: a mixture of restarted e-processes with a time-uniform false-alarm guarantee, a tolerance band that accounts for the uncertainty of its own reference, an additive decomposition that names the cause and the feature, a cohort-aware dashboard with online error control, and a maintenance policy that turns a diagnosis into the cheapest sufficient action. On real employee data the resulting system is the only monitor evaluated that never fires on fairness-irrelevant drift while reliably firing on harmful drift, and it reaches oracle-level disparity control at roughly two percent of the labelling cost of a periodic schedule. We release the implementation and all experiment scripts.

References

[1] Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. In Advances in Neural Information Processing Systems (pp. 3315–3323).

[2] Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. In Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (pp. 214–226).

[3] Chouldechova, A. (2017). Fair prediction with disparate impact: A study of bias in recidivism prediction instruments. Big Data, 5(2), 153–163.

[4] Kleinberg, J., Mullainathan, S., & Raghavan, M. (2017). Inherent trade-offs in the fair determination of risk scores. In Proceedings of the 8th Innovations in Theoretical Computer Science Conference (pp. 43:1–43:23).

[5] Kamiran, F., & Calders, T. (2012). Data preprocessing techniques for classification without discrimination. Knowledge and Information Systems, 33(1), 1–33.

[6] Zafar, M. B., Valera, I., Gomez-Rodriguez, M., & Gummadi, K. P. (2017). Fairness constraints: Mechanisms for fair classification. In Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (pp. 962–970).

[7] Agarwal, A., Beygelzimer, A., Dudik, M., Langford, J., & Wallach, H. (2018). A reductions approach to fair classification. In Proceedings of the 35th International Conference on Machine Learning (pp. 60–69).

[8] Pleiss, G., Raghavan, M., Wu, F., Kleinberg, J., & Weinberger, K. Q. (2017). On fairness and calibration. In Advances in Neural Information Processing Systems (pp. 5680–5689).

[9] Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A survey on bias and fairness in machine learning. ACM Computing Surveys, 54(6), 115:1–115:35.

[10] Bellamy, R. K. E., Dey, K., Hind, M., Hoffman, S. C., Houde, S., Kalra, K., Karmahapatra, P., Martino, R., Mehta, S., Mojsilovic, A., Nagar, S., Ramamurthy, K. N., Richards, J., Saha, D., Sattigeri, P., Smith, C., Varshney, K. R., & Zhang, Y. (2019). AI Fairness 360: An extensible toolkit for detecting and mitigating algorithmic bias. IBM Journal of Research and Development, 63(4/5), 4:1–4:15.

[11] Bird, S., Dudik, M., Edgar, R., Horn, B., Lutz, R., Milan, V., Sameki, M., Wallach, H., & Walker, K. (2020). Fairlearn: A toolkit for assessing and improving fairness in AI. Microsoft Research, Technical Report MSR-TR-2020-32.

[12] Saleiro, P., Kuester, B., Hinkson, L., London, J., Stevens, A., Anisfeld, A., Rodolfa, K. T., & Ghani, R. (2018). Aequitas: A bias and fairness audit toolkit. https://arxiv.org/abs/1811.05577

[13] Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Bass, L., Bateman, C., Brinton, H., Hua, R., Hurst, K., Kim, L., Lee, J., & Viegas, F. (2019). Model cards for model reporting. In Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (pp. 220–229).

[14] Raghavan, M., Barocas, S., Kleinberg, J., & Levy, K. (2020). Mitigating bias in algorithmic hiring: Evaluating claims and practices. In Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (pp. 469–481).

[15] Wilson, C., Ghosh, A., Jiang, S., Mislove, A., Baker, L., Szary, J., Trindel, K., & Polli, F. (2021). Building and auditing fair algorithms: A case study in candidate screening. In Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (pp. 666–677).

[16] Barocas, S., & Selbst, A. D. (2016). Big data's disparate impact. California Law Review, 104(3), 671–732.

[17] Angwin, J., Larson, J., Mattu, S., & Kirchner, L. (2016, May 23). Machine bias. ProPublica.

[18] Gama, J., Zliobaite, I., Bifet, A., Pechenizkiy, M., & Bouchachia, A. (2014). A survey on concept drift adaptation. ACM Computing Surveys, 46(4), 44:1–44:37.

[19] Lu, J., Liu, A., Dong, F., Gu, F., Gama, J., & Zhang, G. (2019). Learning under concept drift: A review. IEEE Transactions on Knowledge and Data Engineering, 31(12), 2346–2363.

[20] Bifet, A., & Gavalda, R. (2007). Learning from time-changing data with adaptive windowing. In Proceedings of the SIAM International Conference on Data Mining (pp. 443–448).

[21] Gama, J., Medas, P., Castillo, G., & Rodrigues, P. (2004). Learning with drift detection. In Advances in Artificial Intelligence (pp. 286–295).

[22] Page, E. S. (1954). Continuous inspection schemes. Biometrika, 41(1/2), 100–115.

[23] Rabanser, S., Guennemann, S., & Lipton, Z. C. (2019). Failing loudly: An empirical study of methods for detecting dataset shift. In Advances in Neural Information Processing Systems (pp. 1396–1408).

[24] Gretton, A., Borgwardt, K. M., Rasch, M. J., Schoelkopf, B., & Smola, A. (2012). A kernel two-sample test. Journal of Machine Learning Research, 13, 723–773.

[25] Lipton, Z. C., Wang, Y.-X., & Smola, A. (2018). Detecting and correcting for label shift with black box predictors. In Proceedings of the 35th International Conference on Machine Learning (pp. 3122–3130).

[26] Ding, F., Hardt, M., Miller, J., & Schmidt, L. (2021). Retiring Adult: New datasets for fair machine learning. In Advances in Neural Information Processing Systems (pp. 6478–6490).

[27] Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Kreuger, J., & Dennison, D. (2015). Hidden technical debt in machine learning systems. In Advances in Neural Information Processing Systems (pp. 2503–2511).

[28] Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2017). The ML test score: A rubric for ML production readiness and technical debt reduction. In Proceedings of the IEEE International Conference on Big Data (pp. 1123–1132).

[29] Rhee, M., Zou, J., Mo, T., Teng, D., & Yang, J.-S. (2026). Enhancing web search agents with self-play contrastive fine-tuning. IEEE Access. https://doi.org/10.1109/ACCESS.2026.3717594

[30] Fan, H., Jiao, Y., Wang, M., Chen, J., & Yang, S. (2026). Fraud learns too: Continual graph learning under strategic adversarial drift in dynamic networks. Scientific Reports. https://doi.org/10.1038/s41598-026-60997-7

[31] D'Amour, A., Srinivasan, H., Atwood, J., Baljekar, P., Sculley, D., & Halpern, Y. (2020). Fairness is not static: Deeper understanding of long term fairness via simulation studies. In Proceedings of the ACM Conference on Fairness, Accountability, and Transparency (pp. 525–534).

[32] Liu, L. T., Dean, S., Rolf, E., Simchowitz, M., & Hardt, M. (2018). Delayed impact of fair machine learning. In Proceedings of the 35th International Conference on Machine Learning (pp. 3150–3158).

[33] Zhang, W., & Ntoutsi, E. (2019). FAHT: An adaptive fairness-aware decision tree classifier. In Proceedings of the 28th International Joint Conference on Artificial Intelligence (pp. 1480–1486).

[34] Iosifidis, V., & Ntoutsi, E. (2020). FABBOO: Online fairness-aware learning under class imbalance. In Discovery Science (pp. 159–174).

[35] Zhang, W., Bifet, A., Zhang, X., Weiss, J. C., & Nejdl, W. (2021). FARF: A fair and adaptive random forests classifier. In Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining (pp. 245–256).

[36] Ghosh, A., Shanbhag, A., & Wilson, C. (2022). FairCanary: Rapid continuous explainable fairness. In Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society (pp. 307–316).

[37] Waudby-Smith, I., & Ramdas, A. (2024). Estimating means of bounded random variables by betting. Journal of the Royal Statistical Society Series B, 86(1), 1–27.

[38] Ramdas, A., Grunwald, P., Vovk, V., & Shafer, G. (2023). Game-theoretic statistics and safe anytime-valid inference. Statistical Science, 38(4), 576–601.

[39] Vovk, V., & Wang, R. (2021). E-values: Calibration, combination and applications. The Annals of Statistics, 49(3), 1736–1754.

[40] Wang, R., & Ramdas, A. (2022). False discovery rate control with e-values. Journal of the Royal Statistical Society Series B, 84(3), 822–852.

[41] Xu, Z., & Ramdas, A. (2024). Online multiple testing with e-values. In Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (pp. 3997–4005).

[42] Shin, J., Ramdas, A., & Rinaldo, A. (2023). E-detectors: A nonparametric framework for sequential change detection. The New England Journal of Statistics in Data Science, 2(2), 229–260.

[43] Chugg, B., Cortes-Gomez, S., Wilder, B., & Ramdas, A. (2023). Auditing fairness by betting. In Advances in Neural Information Processing Systems.

[44] Cherian, J. J., & Candes, E. J. (2024). Statistical inference for fairness auditing. Journal of Machine Learning Research, 25, 1–49.

[45] Maneriker, P., Burley, C., & Parthasarathy, S. (2023). Online fairness auditing through iterative refinement. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 1665–1676).

[46] Chen, Z., Wang, M., Zeng, Z., & Ping, W. (2026). Uncertainty-aware financial forecasting: Leveraging conformal prediction for risk-adjusted models. IEEE Access, 14, 90053–90077. https://doi.org/10.1109/ACCESS.2026.3702666

[47] Liang, Y., Jiao, Y., Ping, W., Fan, H., & Han, X. (2026). Adaptive event-driven labeling: A neuro-symbolic multiagent framework for causal inference in non-stationary time series. IEEE Access, 14, 104468–104482. https://doi.org/10.1109/ACCESS.2026.3709267

[48] European Parliament and Council of the European Union. (2024). Regulation (EU) 2024/1689 laying down harmonised rules on artificial intelligence (Artificial Intelligence Act). Official Journal of the European Union.