Studies in Knowledge and Innovation
Vol. 2 No. 4A (2026)

Buffer Trend Without Predictive Gain in One Day of Streaming Reports

Christopher Sanders[1] · Logan Mason[2] · Ryan King[3]

[1] Lithuania · [2] Iceland · [3] Luxembourg

Published: April 30, 2026 | Article 840

Abstract

Recent buffer movement seems a useful addition to the buffer level itself, but that expectation did not hold in this fixed streaming-log experiment. Two ridge-logistic scores were fitted to 109,254 earlier windows and evaluated on 128,224 later windows from session keys absent before the cutoff. Adding a strictly past buffer slope increased Brier loss by 0.00000651; the increase persisted under every single-hour and single-session deletion. Average precision declined from 0.433924 to 0.430695 against an event share of 0.001474. The response was a rebuffer report in a subsequent server-receipt interval. Complete future reporting was required for scoring, excluding 7,161 otherwise supported windows. The finding concerns incremental prediction within that selected archive, without identifying physical interruption timing, viewing exits or service causes.

Buffer Trend Without Predictive Gain in One Day of Streaming Reports

Abstract

Recent buffer movement seems a useful addition to the buffer level itself, but that expectation did not hold in this fixed streaming-log experiment. Two ridge-logistic scores were fitted to 109,254 earlier windows and evaluated on 128,224 later windows from session keys absent before the cutoff. Adding a strictly past buffer slope increased Brier loss by 0.00000651; the increase persisted under every single-hour and single-session deletion. Average precision declined from 0.433924 to 0.430695 against an event share of 0.001474. The response was a rebuffer report in a subsequent server-receipt interval. Complete future reporting was required for scoring, excluding 7,161 otherwise supported windows. The finding concerns incremental prediction within that selected archive, without identifying physical interruption timing, viewing exits or service causes.

1. A changing state is not automatically a better predictor

A player with little buffered video has limited recorded playback reserve. A declining buffer may appear to add an obvious warning: the reserve is being depleted. Yet the value of that extra measurement is an empirical question. Buffer level and recent change overlap, adaptations react to the same conditions, and a short trend can reflect the timing of chunk arrivals or client reports. A predictor can therefore contain more history without improving its later probability scores.

We ask whether the preceding five-second buffer trend adds predictive value beyond the current reported level for a rebuffer report received during the following thirty seconds. We then assess the addition in later session keys that were absent from earlier source records. The comparison uses identical eligible windows for the two scores. It neither assigns a cause to a rebuffer report nor estimates what would happen if an operator artificially changed a player's buffer.

Buffer-based adaptation provides a substantive reason to inspect occupancy: BOLA uses buffer information in its control policy (Spiteri et al., 2016). Model-predictive streaming instead places forecasts within a control objective (Yin et al., 2015), while probe-and-adapt designs respond to changing delivery conditions (Li et al., 2014). These are distinct mechanisms, none rerun here. They also explain why a measured buffer is partly an outcome of operating policies, rather than an externally assigned treatment.

Loading diagnosis requires additional evidence. A working-paper abstract describes dynamic service-dependency graphs, delay prediction and cause localization in media workflows (Yuan et al., 2026). Our endpoint reports contain no corresponding dependency graph or adjudicated service failure. The relevant connection is the boundary between recognizing a playback precursor and localizing a service cause. Our study can examine the first problem under a specific recording contract; it cannot validate the working paper's localization results or underlying platform data.

The distinction between explanation and prediction matters here (Shmueli, 2010). Even a successful warning would not identify a network bottleneck, cache failure or viewer preference. Moreover, streaming experience has several dimensions: HTTP measurements examine multiple quality contributors (Mok et al., 2011), and adaptive-streaming surveys organize a broader quality-of-experience problem (Seufert et al., 2015). The single response used below represents none of those dimensions in their entirety.

2. What the timestamp and event actually represent

The source is Puffer's official public client_buffer release for 12 November 2026 at 11:00 UTC through 13 November at 11:00 UTC (Stanford Puffer project, n.d.-a). We retained one fixed day rather than searching dates for favorable results. The Puffer research platform supplies real streaming measurements under variable operating conditions (Yan et al., 2020); this reanalysis does not reproduce its experimental comparisons or rank the streaming schemes in the later release.

Actual source columns contain a nanosecond timestamp, a public session identifier, stream index, experiment identifier, content channel, event, buffer and cumulative rebuffer time. The observed event categories are init, startup, timer, rebuffer and play. A session starts with a page load or reload in the documented grouping, while channel changes create streams within a session. No explicit voluntary-exit or swipe field is established by these records. We discard the content channel and retain only technical fields with locally assigned session equality keys.

Crucially, the documented timestamp is when the server receives a client message. The pinned release code preserves the database timestamp and writes it to the time(ns GMT) column (Stanford Puffer project, n.d.-b). It loops through server and channel tables before their timestamp maps, so global row order need not follow time. The complete file has 342 global time reversals, although each stream's timestamps are strictly increasing with no ties. Window construction follows each stream's own temporal history.

The outcome therefore means a rebuffer **report** in the following server-receipt interval. It is not an independently synchronized physical onset. Buffer values and cumulative counters are also reported quantities; their client-to-server delay is not reconstructed. Measurement distinctions of this kind determine what construct an observed number can support (Jacobs and Wallach, 2021). They cannot be repaired by describing a score as real-time simply because its features precede its recorded response.

The original object contains 231,204,593 gzip bytes and expands to 3,851,515,693 bytes. We verified its storage generation, complete compressed length, provider MD5, original-byte SHA256, and gzip CRC and size trailer. Streaming projection produced 40,456,730 rows, 3,267 session keys and 22,888 stream keys. There are 2,062 rebuffer-report rows. No within-stream cumulative-counter decrease, repeated init or repeated startup was found. Only minimal technical projections and provenance remain after disposable original-byte working files were removed.

3. Future observation defines the scored cohort

Source and window feasibility were inspected before model fitting. The final protocol was frozen after response-support counts were known, making this a secondary archived-data experiment rather than a preregistered or outcome-blinded study. We required an observed init at or before startup. A timer becomes a candidate only when the most recent named state is playing, established by startup or play rather than rebuffer. Thus the analysis does not intentionally predict continuation of an already reported interruption.

Candidates begin sixty seconds after startup. Each selected timer sets the earliest next anchor thirty seconds later, before its support is assessed. An anchor must arrive within two seconds of its scheduled time. The current buffer is measured at that timer. The earlier value is the most recent report at or before five seconds earlier; the slope divides their buffer difference by the actual elapsed seconds. Every reporting gap across that past interval must be at most two seconds. No future report contributes to either feature.

A future window is supported only when a report spans the thirty-second endpoint, arrives no more than two seconds beyond it, and leaves no intervening reporting gap above two seconds. The binary response is one when an explicit rebuffer report lies in the open-closed interval from anchor to anchor plus thirty seconds. A report after that endpoint can establish coverage but cannot create a positive response inside the horizon. Unsupported responses remain missing. Coverage is determined retrospectively and never supplied as a predictor.

These rules yield 299,789 candidates. Of these, 298,610 have acceptable anchor timing and past support; 291,449 also have complete future support. Exactly 335 supported windows contain a rebuffer report. This is a count of positive windows, distinct from the 2,062 raw report rows across every playback phase. Streams without an observed init or startup, reports before the burn-in, and windows failing the support rules do not enter the same denominator. We do not invent events to connect those different denominators.

Among the 7,161 otherwise supported windows without full future coverage, 137 already contain a rebuffer report in their available future prefix. Their response remains unscored under the declared common-coverage rule. Treating them as negative would erase observed information. Missing-data analysis requires assumptions about the observation process (Little and Rubin, 2002); selection can also change the relationships in a fitted sample (Heckman, 1979). We do not fit a selection correction or assume the omitted windows are exchangeable with the scored ones.

4. A small comparison with an explicit temporal split

The fixed split is 23:00 UTC, halfway through the release. Fitting windows require their verified future endpoint to precede this boundary. There are 109,254 such windows, including 115 positive windows from 41 sessions. Another 128 fully supported windows span the boundary and do not enter fitting. This prevents a nominally early anchor from bringing a later outcome into the training period.

The principal later cohort contains 128,224 windows from 1,070 session keys whose first observation anywhere in the complete source occurs at or after the cutoff. It has 189 positive windows from 42 sessions and no session-key intersection with fitting. This establishes absence from the captured earlier source, not a new person, a newly created real-world session, or independence from every earlier viewer. The secondary later cohort contains 53,843 windows from continuing session keys, with only 31 positives from six sessions.

We fit a logistic score using log(1 + current buffer), then another adding the signed past buffer slope. Centering and scaling use fitting windows only. Both minimize mean binomial negative log likelihood plus one-half of 0.001 times the squared standardized non-intercept coefficients. The quadratic regularization principle follows ridge regression (Hoerl and Kennard, 1970); its use here is a penalized predictive logistic procedure (Hastie et al., 2009). The intercept is unpenalized, negative slopes are retained, and no feature, penalty or horizon is selected using later performance.

Both fits converge in twelve or thirteen iterations, with maximum absolute gradients below 0.000000009. The level-only standardized coefficient is -0.482765. In the augmented score, the level and trend coefficients are -0.475968 and -0.100542. These are scoring parameters conditional on the fitting cohort, not causal effects or uncertainty intervals. A constant reference uses the training event prevalence with one positive and one negative pseudocount. A zero-probability reference was added after initial scoring at an explicit review request; the original fitted comparison was unchanged.

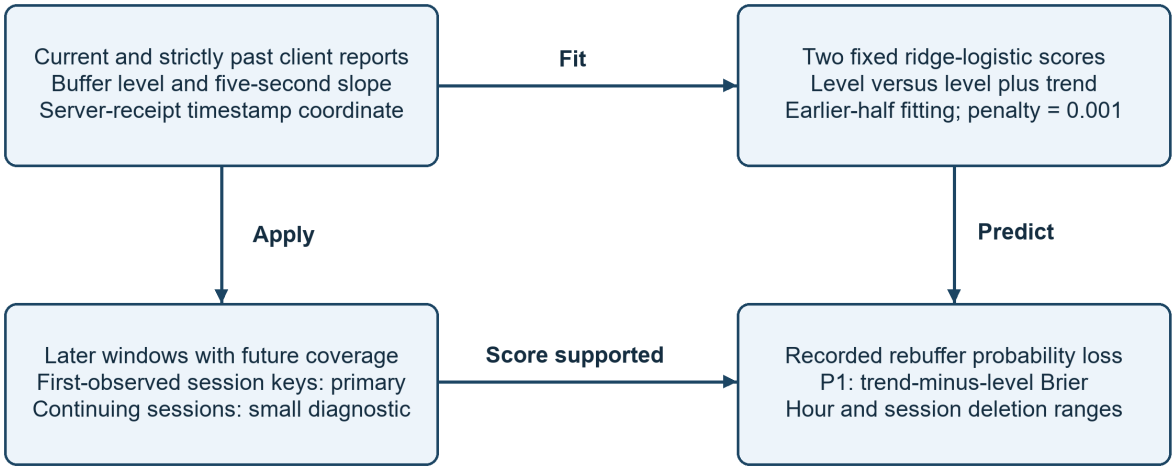


Figure 1. Research model. *Current buffer and strictly past trend feed two fixed scoring specifications. Later windows are separated by captured session history. Future report support is a retrospective scoring condition, not a feature known at prediction time. The primary comparison concerns received-report probability loss.*

5. More history produced a small adverse increment

Proposition P1 concerns the augmented-score Brier loss minus level-only Brier loss in the principal later cohort. Brier's squared probability-error score defines the calculation (Brier, 1950), and proper scoring rules motivate checking probabilities without choosing a classification threshold (Gneiting and Raftery, 2007). A positive increment means that adding trend worsened agreement with the later recorded response on the same windows.

Table 1. Paired changes from adding the strictly past buffer slope.

Evaluation cohort	Windows	Positive windows	Sessions / positive sessions	ΔBrier × 10^6	Delete-hour range × 10^6	Delete-session range × 10^6
Later first-observed sessions	128224	189	1070 / 42	6.5081	[4.3398, 9.2185]	[2.2454, 7.9000]
Later continuing sessions	53843	31	136 / 6	6.2328	[3.5800, 7.9184]	[1.8461, 6.4307]

Both ridge-logistic specifications use 109,254 fitting windows and penalty 0.001. Every effect and deletion-range value in Table 1 is multiplied by one million. Models remain fixed while each later UTC hour or session key is omitted. The ranges describe finite deletion sensitivity, not confidence intervals, p-values or independent replications. Outcomes are rebuffer-report windows with full reporting support.

In the principal cohort, Brier loss increases from 0.00112198 to 0.00112848, a difference of 0.0000065081. The increment remains positive after any one of the twelve evaluation hours or any one of the 1,070 evaluation sessions is omitted. The level-only score therefore performs slightly better under this fixed specification. The result does not establish that buffer trend has no information under every representation, interval, policy or operating day.

The continuing-session diagnostic has the same direction, with a difference of 0.0000062328. Its many windows should not obscure the six sessions contributing positive responses. This is a small supporting

cohort, not an independent platform replication or a precise comparison between continuing and first-observed viewers. We do not interpret the difference between its increment and the principal increment as a session-history effect.

Table 2. Probability and ranking performance on the identical scored windows.

Cohort	Score	Brier	Log loss	AUROC	Average precision
Later first-observed sessions	Zero probability	0.00147398	∞	0.500000	0.001474
Later first-observed sessions	Training prevalence	0.00147198	0.01115435	0.500000	0.001474
Later first-observed sessions	Buffer level	0.00112198	0.00520192	0.976347	0.433924
Later first-observed sessions	Level + trend	0.00112848	0.00523790	0.968401	0.430695
Later continuing sessions	Zero probability	0.00057575	∞	0.500000	0.000576
Later continuing sessions	Training prevalence	0.00057565	0.00500432	0.500000	0.000576
Later continuing sessions	Buffer level	0.00047440	0.00294959	0.981779	0.340820
Later continuing sessions	Level + trend	0.00048063	0.00297415	0.953631	0.339989

Table 2 reports 128,224 principal or 53,843 continuing-session windows. Zero probability assigns every window probability zero; positive responses make its log loss infinite. Saved machine-readable results retain a missing numeric cell plus an explicit infinity status, rather than replace it with a finite value. The constant reference uses only training prevalence. No thresholded accuracy is used to judge the added feature.

6. Rare events make the reference score consequential

The principal event share is 0.001473983, or approximately 0.1474%. Its zero-probability Brier loss is exactly that share, and the training-prevalence reference scores 0.00147198. The level score has lower Brier loss than either reference. Its average precision of 0.433924 also exceeds the constant-score reference of 0.001474. This contextualizes its ranking without interpreting precision as proof of physical interruption or a deployable alert policy.

AUROC summarizes ranking (Fawcett, 2006), but classifier-performance critiques caution against allowing one ranking summary to determine every conclusion (Hand, 2009). Precision-recall analysis asks a different question and depends on the positive class frequency (Davis and Goadrich, 2006). Here adding trend reduces principal AUROC from 0.976347 to 0.968401 and average precision from 0.433924 to 0.430695. Log loss rises from 0.00520192 to 0.00523790. These secondary measures agree with the small adverse primary increment.

The observation contract remains central despite these favorable level-only scores. Supported windows have a mean current buffer of 15.4346 seconds; otherwise eligible unsupported windows average 15.0186 seconds. Their mean prior slopes also differ. These descriptive differences do not identify a missingness mechanism, but they show why performance conditional on complete future reporting cannot

automatically represent all attempted playback. Reporting cessation may reflect gaps, a switch, the day's boundary or an unknown ending; it is not a verified voluntary exit.

Research connecting stream quality with viewer behavior uses additional identification evidence (Krishnan and Sitaraman, 2013). Our archive contains neither such a design nor an observed swipe. We therefore do not translate the prediction into abandonment, satisfaction or a preference signal. A strong association with a received technical report is useful only within that measurement scope.

7. What the negative addition does and does not settle

An honest temporal split is necessary but has a defined reach. Cross-validation reviews distinguish fitting choices from performance estimation (Arlot and Celisse, 2010); validation with structured observations depends on the groups and times withheld (Roberts et al., 2017). Leakage analysis likewise requires an explicit information boundary (Kapoor and Narayanan, 2023). Our split isolates earlier source keys from the principal later cohort and excludes boundary-spanning outcomes. It does not reproduce new days, independent people, different platforms or synchronized client-side clocks.

Repeated windows still share streams and sessions. Grouped inference depends on the relevance of the grouping unit (Cameron and Miller, 2015), and a public session key is not a person identifier. Our deletions hold predictions fixed and inspect sensitivity to captured groups. Deletion diagnostics have a broader statistical history (Cook, 1977), but these ranges are not Cook's distance, model refits or population uncertainty. The fact that every deletion preserves the direction does not remove unmeasured reporting delays or selection.

The experiment also pools seven recorded scheme identifiers without using them as features. Current level and trend can reflect control responses, encoding conditions and delivery variation. Content-channel information was deliberately discarded. The fitted increment is therefore a comparison between two specified summaries in one observed operating mixture. It does not reveal why trend failed to help, nor whether a different control policy would change the result.

Collection documentation should expose actual fields, intended use and observation limits (Gebru et al., 2021). Separating raw event rows, stream histories and scored windows keeps their different analytical units visible (Wickham, 2014). Computational reproducibility requires inspectable transformations and saved numerical objects (Peng, 2011), together with retained intermediate decisions (Sandve et al., 2013). We preserve source hashes, window eligibility, missing responses, coefficients and all later probabilities, and independently verify selected windows against the technical event projection. These checks support the calculation, not an unmeasured social or physical outcome.

8. Conclusion

Current reported buffer level supplied useful conditional discrimination in this fixed day, while adding a short past trend slightly worsened later probability scores. The adverse increment persisted across the declared hour and session deletions, including in the principal source-key-isolated cohort. The finding is narrower than a general rejection of history: it tests two fixed, regularized summaries against received rebuffer reports. Complete-future support, server-clock timing and rare positive sessions delimit the result. They also explain why a playback warning cannot be treated as an identified root cause or a measured viewing exit.

References

1. Fawcett, Tom. (2006). An introduction to ROC analysis. Pattern Recognition Letters. https://doi.org/10.1016/j.patrec.2005.10.010.

2. Cook, R. Dennis. (1977). Detection of Influential Observation in Linear Regression. Technometrics. https://doi.org/10.1080/00401706.1977.10489493.

3. Hand, David J.. (2009). Measuring classifier performance: a coherent alternative to the area under the ROC curve. Machine Learning. https://doi.org/10.1007/s10994-009-5119-5

4. Gneiting, Tilmann; Raftery, Adrian E. (2007). Strictly Proper Scoring Rules, Prediction, and Estimation. Journal of the American Statistical Association. https://doi.org/10.1198/016214506000001437.

5. Brier, Glenn W.. (1950). Verification of forecasts expressed in terms of probability. Monthly Weather Review. https://doi.org/10.1175/1520-0493(1950)078<0001:vofeit>2.0.co;2

6. Sandve, Geir Kjetil; Nekrutenko, Anton; Taylor, James; Hovig, Eivind. (2013). Ten Simple Rules for Reproducible Computational Research. PLoS Computational Biology. https://doi.org/10.1371/journal.pcbi.1003285

7. Yuan, Yuewei; Xu, Tianyi; Yin, Jiayang; Huang, Jialei. (2026). Modeling Loading Anomalies and Identifying Root Causes in Media Consumption Workflows on Large Social Media Platforms. SSRN working paper 7393318. https://doi.org/10.2139/ssrn.7393318.

8. Seufert, Michael; Egger, Sebastian; Slanina, Martin; Zinner, Thomas; Hoßfeld, Tobias; Tran-Gia, Phuoc. (2015). A Survey on Quality of Experience of HTTP Adaptive Streaming. IEEE Communications Surveys & Tutorials. https://doi.org/10.1109/comst.2014.2360940

9. Shmueli, Galit. (2010). To Explain or to Predict?. Statistical Science. https://doi.org/10.1214/10-sts330

10. Yan, Francis Y.; Ayers, Hudson; Zhu, Chenzhi; Fouladi, Sadjad; Hong, James; Zhang, Keyi; Levis, Philip; Winstein, Keith. (2020). Learning in situ: a randomized experiment in video streaming. 17th USENIX Symposium on Networked Systems Design and Implementation. https://www.usenix.org/conference/nsdi20/presentation/yan

11. Stanford Puffer project. (n.d.-b). Puffer-statistics: public release pipeline, pinned source revision 59afb3149b6249dcbedca4cfd9dcd7025ed8fd00. Accessed 16 November 2026. https://github.com/StanfordSNR/puffer-statistics/tree/59afb3149b6249dcbedca4cfd9dcd7025ed8fd00

12. Little, Roderick J. A.; Rubin, Donald B.. (2002). Statistical Analysis with Missing Data. Wiley Series in Probability and Statistics. https://doi.org/10.1002/9781119013563

13. Wickham, Hadley. (2014). Tidy Data. Journal of Statistical Software. https://doi.org/10.18637/jss.v059.i10

14. Li, Zhi; Zhu, Xiaoqing; Gahm, Joshua; Pan, Rong; Hu, Hao; Begen, Ali C.; Oran, David. (2014). Probe and Adapt: Rate Adaptation for HTTP Video Streaming At Scale. IEEE Journal on Selected Areas in Communications. https://doi.org/10.1109/jsac.2014.140405

15. Kapoor, Sayash; Narayanan, Arvind. (2023). Leakage and the reproducibility crisis in machine-learning-based science. Patterns. https://doi.org/10.1016/j.patter.2023.100804

16. Davis, Jesse; Goadrich, Mark. (2006). The relationship between Precision-Recall and ROC curves. Proceedings of the 23rd international conference on Machine learning - ICML '06. https://doi.org/10.1145/1143844.1143874

17. Hastie, Trevor; Tibshirani, Robert; Friedman, Jerome. (2009). The Elements of Statistical Learning. Springer Series in Statistics. https://doi.org/10.1007/978-0-387-84858-7

18. Cameron, A. Colin, and Douglas L. Miller. “A Practitioner’s Guide to Cluster-Robust Inference.” *Journal of Human Resources*, vol. 50, no. 2, 2015, pp. 317-372. https://doi.org/10.3368/jhr.50.2.317.

19. Gebru, Timnit, et al. "Datasheets for datasets" *Communications of the ACM*, 2021. https://doi.org/10.1145/3458723.

20. Jacobs, Abigail Z.; Wallach, Hanna. (2021). Measurement and Fairness. Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency. https://doi.org/10.1145/3442188.3445901

21. Krishnan, S. Shunmuga; Sitaraman, Ramesh K.. (2013). Video Stream Quality Impacts Viewer Behavior: Inferring Causality Using Quasi-Experimental Designs. IEEE/ACM Transactions on Networking. https://doi.org/10.1109/tnet.2013.2281542

22. Roberts, David R., et al. “Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure.” *Ecography*, vol. 40, no. 8, 2017, pp. 913-929. https://doi.org/10.1111/ecog.02881.

23. Yin, Xiaoqi; Jindal, Abhishek; Sekar, Vyas; Sinopoli, Bruno. (2015). A Control-Theoretic Approach for Dynamic Adaptive Video Streaming over HTTP. Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication. https://doi.org/10.1145/2785956.2787486

24. Peng, Roger D.. (2011). Reproducible Research in Computational Science. Science. https://doi.org/10.1126/science.1213847

25. Mok, Ricky K. P.; Chan, Edmond W. W.; Chang, Rocky K. C.. (2011). Measuring the quality of experience of HTTP video streaming. 12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops. https://doi.org/10.1109/inm.2011.5990550

26. Stanford Puffer project. (n.d.-a). Puffer: public streaming measurements and data description. Accessed 16 November 2026. https://puffer.stanford.edu/data-description/

27. Heckman, James J.. (1979). Sample Selection Bias as a Specification Error. Econometrica. https://doi.org/10.2307/1912352

28. Spiteri, Kevin; Urgaonkar, Rahul; Sitaraman, Ramesh K.. (2016). BOLA: Near-optimal bitrate adaptation for online videos. IEEE INFOCOM 2016 - The 35th Annual IEEE International Conference on Computer Communications. https://doi.org/10.1109/infocom.2016.7524428

29. Hoerl, Arthur E.; Kennard, Robert W.. (1970). Ridge Regression: Biased Estimation for Nonorthogonal Problems. Technometrics. https://doi.org/10.1080/00401706.1970.10488634.

30. Arlot, Sylvain; Celisse, Alain. (2010). A survey of cross-validation procedures for model selection. Statistics Surveys. https://doi.org/10.1214/09-ss054