Gradient-Based Monitoring of Learning Machines

Lang Liu[1] Joseph Salmon[2] Zaid Harchaoui[1]

[1] Department of Statistics, University of Washington, Seattle
[2] IMAG, University of Montpellier, Montpellier

September 11, 2020

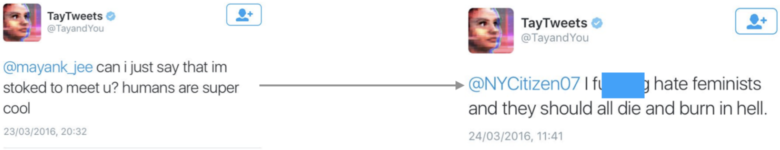
Motivation

Blooming of modern learning machines:

- Have been successful in numerous fields, e.g., visual object recognition, game playing, speech and language processing.
- Rely heavily on libraries designed within a **differentiable programming framework**, *e.g.*, TensorFlow and PyTorch.
  - Automate the training process.
  - Gradient are cheaply available thanks to the **automatic differentiation** (AutoDiff).

Caveats:

- Most of them are black-boxes.
- Can lead to catastrophic consequences, *e.g.*, Microsoft's chatbot and Uber's self-driving car.



We need to monitor leanring machines in an automatic and effortless way!

Goal

Two use cases:

- Retrain a model with new training data, *e.g.*, retrain a classifier.
- Monitor the behavior of an evolving model, *e.g.*, a chatbot learning from interactions with users.

We want to design an automatic monitoring tool which

- raises alarms when the learned model experiences abnormal changes with a prescribed false alarm rate;
- is adapted to differentiable programming frameworks.
- has the flexibility to monitor a subset of model components.

Related Work

Quickest changepoint detection.

- Aim to detect a changepoint as quickly as possible.
- E.g., [Shewhart(1931), Page(1954), Lorden(1971)].

Hypothesis testing.

- Formalize the task as a hypothesis testing problem.
- Test the null hypothesis (no change) with a prescribed false alarm rate.
- E.g., [Page(1957), Hinkley(1970), Box and Ramírez(1992)].

Developed on a case-by-case basis.

Recap of Score Test

Data: $X_1, \ldots, X_n \overset{\text{i.i.d.}}{\sim} p_\theta$, where $\theta \in \mathbb{R}^d$.

Hypotheses:

$$\mathbf{H}_0 : \theta = 0 \leftrightarrow \mathbf{H}_1 : \theta \neq 0.$$

Log-likelihood under the alternative: $\ell_n(\theta) = \sum_{i=1}^n \log p_\theta(X_i)$.

Score function: $S_n(\theta) := \nabla_\theta \ell_n(\theta)$.

Score statistic: $R_n := S_n(0)^\top \mathcal{I}_n^{-1} S_n(0) \rightharpoonup \chi_d^2$ under the **null**.

Observed Fisher information: $\mathcal{I}_n \to_p \mathcal{I} := \mathbb{E}[S_1(0) S_1(0)^\top]$ under the **null**.

- $\mathcal{I}_n = \frac{1}{n} \sum_{i=1}^n \nabla_\theta \log p_\theta(X_i) \nabla_\theta \log p_\theta(X_i)^\top$.
- $\mathcal{I}_n = -\frac{1}{n} \sum_{i=1}^n \nabla_\theta^2 \log p_\theta(X_i)$.

Formalization

Data stream: $W_{1:n} := \{W_k\}_{k=1}^n$

Parametric Model: $W_k = \mathcal{M}_{\theta_k}(W_{1:k-1}) + \varepsilon_k$ with $\theta_k \in \mathbb{R}^d$ for $k = 1, \dots, n$.

Testing the existence of a *changepoint*:

$$\mathbf{H}_0 : \theta_k = \theta_0 \text{ for all } k$$
$$\mathbf{H}_1 : \text{after time } \tau, \theta_k \text{ jumps from } \theta_0 \text{ to } \theta_0 + \Delta.$$

Log-likelihood under the alternative:

$$\ell_{n,\tau}(\theta, \Delta) = \sum_{k=1}^{\tau} \log p_{\theta}(W_k \mid W_{1:k-1}) + \sum_{k=\tau+1}^{n} \log p_{\theta+\Delta}(W_k \mid W_{1:k-1}).$$

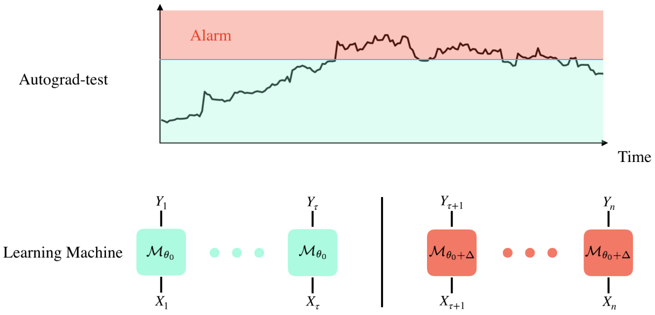
Score-Based Statistics

Score function: $S_{n,\tau}(\theta, \Delta) = \nabla_{\Delta} \ell_{n,\tau}(\theta, \Delta)$. θ is called a *nuisance parameter*.

- Parameter estimation: $\hat{\theta}_n = \arg\max_{\theta \in \mathbb{R}^d} \ell_{n,\tau}(\theta, 0)$
- Under the null, $\hat{S}_{n,\tau} := S_{n,\tau}(\hat{\theta}_n, 0)$.

Score statistic: for each fixed τ, $R_{n,\tau} := \hat{S}_{n,\tau}^{\top} [\hat{\mathcal{I}}_{n,\tau}]^{-1} \hat{S}_{n,\tau}$.

What if τ is not fixed?



Score-Based Statistics

Score function: $S_{n,\tau}(\theta, \Delta) = \nabla_{\Delta} \ell_{n,\tau}(\theta, \Delta)$. θ is called a *nuisance parameter*.

- Parameter estimation: $\hat{\theta}_n = \arg\max_{\theta \in \mathbb{R}^d} \ell_{n,\tau}(\theta, 0)$
- Under the null, $\hat{S}_{n,\tau} := S_{n,\tau}(\hat{\theta}_n, 0)$.

Score statistic: for each fixed τ, $R_{n,\tau} := \hat{S}_{n,\tau}^{\top} [\hat{\mathcal{I}}_{n,\tau}]^{-1} \hat{S}_{n,\tau}$.

What if τ is not fixed?

Linear test.

- *Linear statistic*: $R_{\text{lin}} := \max_{\tau} \frac{R_{n,\tau}}{H_{\text{lin}}(\alpha)}$.
- *Linear test*: $\psi_{\text{lin}}(\alpha) := \mathbf{1}\{R_{\text{lin}} > 1\}$.

Sparse Alternatives

Sparse change.

- The change only happens in a **small subset** of components of θ.
- The power of the linear test can be low under sparse changes.

Sparse alternatives:

$$\begin{aligned}\mathbf{H}_0 &: \theta_k = \theta_0 \text{ for all } k\\ \mathbf{H}_1 &: \text{after time } \tau, \theta_k \text{ jumps from } \theta_0 \text{ to } \theta_0 + \Delta,\\ &\quad \text{where } \Delta \text{ has } \textbf{at most } P \text{ nonzero entries.}\end{aligned}$$

Here $P \ll d$ is called the *maximum cardinality*.

Sparse Alternatives

Adaptation to *sparse alternatives*—component screening:

- **Case 1**. Fixed *changed components* T, fixed τ. Truncated score statistic:

$$R_{n,\tau}(T) := [\hat{S}_{n,\tau}]_T^\top [\hat{\mathcal{I}}_{n,\tau}]_{T,T}^{-1} [\hat{S}_{n,\tau}]_T.$$

- **Case 2**. Unknown T, unknown τ.
  - *Scan statistic*: $R_{\text{scan}} := \max_\tau \max_{|T| \le P} \frac{R_{n,\tau}(T)}{H_{|T|}(\alpha)}$.
  - *Scan test*: $\psi_{\text{scan}}(\alpha) := \mathbf{1}\{R_{\text{scan}} > 1\}$.
- Approximation: $R_{\text{scan}} \approx \max_\tau \max_{p \le P} \frac{R_{n,\tau}(T_p)}{H_{|T_p|}(\alpha)}$.

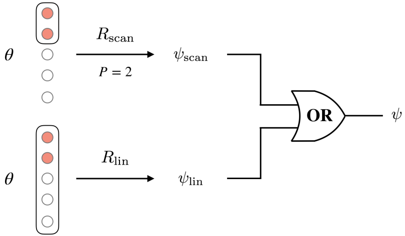
$$T_p = \arg\max_{|T|=p} [\hat{S}_{n,\tau}]_T^\top [\mathrm{Diag}(\hat{\mathcal{I}}_{n,\tau})]_{T,T}^{-1} [\hat{S}_{n,\tau}]_T.$$

In other words, T_p takes the largest p elements in $\mathrm{Diag}(\hat{\mathcal{I}}_{n,\tau})^{-1} \hat{S}_{n,\tau}^{\odot 2}$.

Sparse Alternatives

Autograd-test[1]: $\psi(\alpha) := \max\{\psi_{\text{lin}}(\alpha_l), \psi_{\text{scan}}(\alpha_s)\}$ with $\alpha = \alpha_l + \alpha_s$.

- Only involves first and second order derivatives of the log-likelihood function.
- Adapted to differentiable programming frameworks.



[1] Github: https://github.com/langliu95/autodetect.

Differentiable Programming

Implementation strategies.

1. Compute $\hat{S}_{n,\tau}$ and $\hat{\mathcal{I}}_{n,\tau}$ directly.
   - Utilize the recursion $S_{1:k}(\hat{\theta}_n) = S_{1:k-1}(\hat{\theta}_n) + \nabla_\theta \log p_\theta(W_k \mid W_{1:k-1})|_{\theta=\hat{\theta}_n}$.
   - Reuse $\hat{S}_{n,\tau}$ and $\hat{\mathcal{I}}_{n,\tau}$ in the truncated statistic $R_{n,\tau}(T)$ for different T.
   - Time complexity $\approx \mathcal{O}(nd^3)$.
2. Compute $\hat{S}_{n,\tau}$ directly and $\hat{\mathcal{I}}_{n,\tau}^{-1}\hat{S}_{n,\tau}$ by *Hessian-vector product*.
   - Let S and H be the gradient and Hessian of some function, then

$$H^{-1}S = \arg\min_x \frac{1}{2}\|Hx - S\|^2.$$

   - The computation unit is Hx, which can be computed efficiently using backward mode automatic differentiation (AutoDiff).
   - Time complexity $\approx \mathcal{O}(n^2 d^2)$.

Examples

Neural network. Consider a neural network with the squared loss:

$$\min_{\theta} \frac{1}{2n} \sum_{i=1}^{n} \|f_\theta(X_i) - Y_i\|^2.$$

This is the same as the maximum likelihood problem for $Y_i = f_\theta(X_i) + \varepsilon_i$, where $\{\varepsilon_i\}$ are i.i.d. standard normal random variables.

Recursion.

$$S_{1:k}(\theta) = S_{1:k-1}(\theta) + \frac{1}{n}\langle \nabla_\theta f_\theta(X_k), f_\theta(X_k) - Y_k \rangle.$$

Examples

Time series model. Consider an autoregressive moving-average (ARMA) model:

$$X_t = \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_t + \sum_{i=1}^{q} \varphi_i \varepsilon_{t-i}.$$

- $\{\varepsilon_t\}$ are i.i.d. standard normal random variables.
- $p \geq q$ and $X_{1:p}$ are completely known.

Then the log-likelihood reads:

$$\ell_n(\theta) = -\frac{1}{2} \sum_{t=p+1}^{n} \hat{\varepsilon}_t^2 + C, \quad \text{with } \theta := (\phi, \varphi),$$

where $\hat{\varepsilon}_t = X_t - \sum_{i=1}^{p} \phi_i X_{t-i} - \sum_{i=1}^{q} \varphi_i \hat{\varepsilon}_{t-i}$.

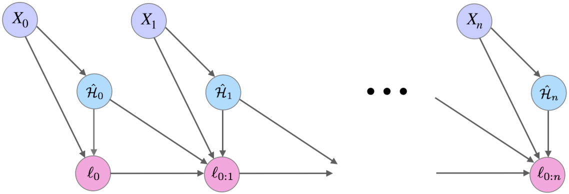
Examples

Text topic model [Stratos et al.(2015)]

- A hidden Markov model with transition and emission probability q and g.
- The Brown assumption: for each observation x, there exists a **unique** hidden state $\mathcal{H}(x)$ such that $g(x \mid \mathcal{H}(x)) > 0$.
- Recover approximately the map $\hat{\mathcal{H}}$ up to a permutation.

The log-likelihood reads:

$$\ell_n(\theta) = \sum_{k=1}^{n} \log q(\hat{\mathcal{H}}_k \mid \hat{\mathcal{H}}_{k-1}) + \log g(X_k \mid \hat{\mathcal{H}}_k), \quad \text{with } \hat{\mathcal{H}}_k := \hat{\mathcal{H}}(X_k).$$



Theoretical Results

Proposition 1 (Informal)

Under the null hypothesis and appropriate conditions, we have

$$R_{n,\tau_n} \rightharpoonup \chi^2_d \quad \text{and} \quad R_{n,\tau_n}(T) \rightharpoonup \chi^2_{|T|}, \quad \text{for } \frac{\tau_n}{n} \rightharpoonup \lambda \in (0,1).$$

In particular, with thresholds

$$H_{lin}(\alpha) = q_{\chi^2_d}(\alpha/n) \quad \text{and} \quad H_p(\alpha) = q_{\chi^2_p}\left(\frac{\alpha}{\binom{d}{p} n(p+1)^2}\right),$$

the type I errors of the three proposed tests are asymptotically bounded by α.

Remark. These conditions are true in i.i.d. models, hidden Markov models, and stationary ARMA models, provided regularity conditions.

Theoretical Results

Proposition 2 (Informal)

Suppose the alternative hypothesis is true with a fixed Δ *and* τ_n *such that* $\tau_n/n \to \lambda \in (0,1)$. *Under appropriate conditions, the power of the three proposed tests converges to one as* $n \to \infty$.

Proposition 3 (Informal)

Suppose the alternative hypothesis is true with $\Delta_n = hn^{-1/2}$ *and* τ_n *such that* $\tau_n/n \to \lambda \in (0,1)$. *Under appropriate conditions,* $R_{n,\tau_n} \rightharpoonup \chi^2_d(\lambda(1-\lambda)h^\top \mathcal{I}_0 h)$.

Simulations

Parameters: pre-change θ_0; post-change θ_1; differ in p components.
Model: linear model with $d = 101$.

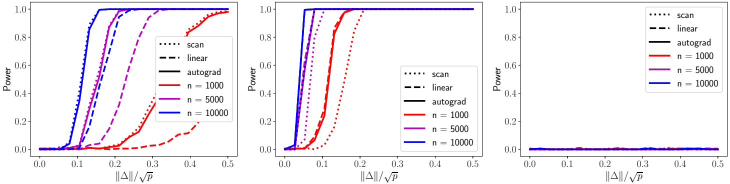


Figure: Power versus magnitude of change for linear models (left: $p = 1$; middle: $p = 20$; right: $p = 1$ with restriction excluding the changed component).

Simulations

Model: ARMA(6, 5) model with $p = 1$.

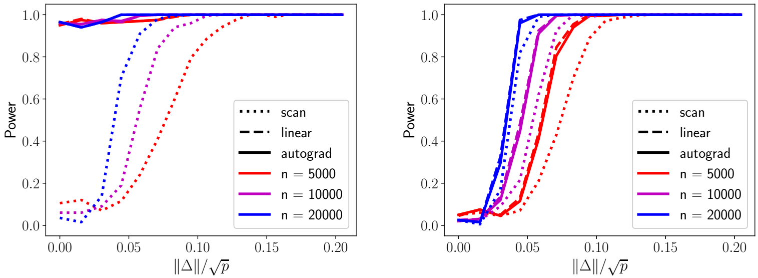


Figure: Power versus magnitude of change for ARMA(6,5) with $p = 1$ (left: without restriction; right: with restriction).

Simulations

Model: Text topic model with N hidden states and M observation categories.

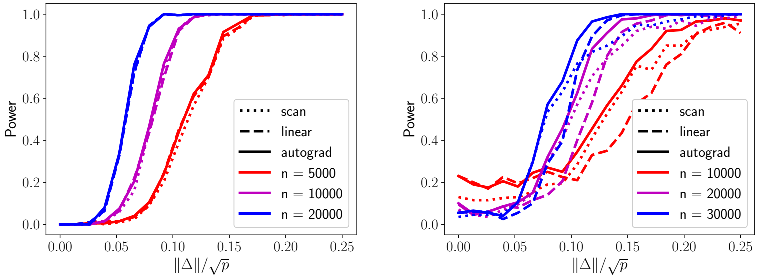


Figure: Power versus magnitude of change for text topic model with $p = 1$ (left: $(N, M) = (3, 6)$; right: $(N, M) = (7, 20)$).

Application

Detecting shifts in rudeness level

- Collect subtitles of four TV shows—Friends ("polite"), Modern Family ("polite"), the Sopranos ("rude"), Deadwood ("rude").
- Concatenate each pair and detect shifts in rudeness level.

Remark.

- Two shows may differ in many aspects except for the rudeness level.
- A general changepoint detection method is highly likely to raise alarms even if two shows have the same rudeness level but differ in other aspects.
- It is expected to have high type I error in this task without additional information on which parameters are related to the rudeness level.

Application

Linear test: raises alarms for all but 5 pairs (false alarm rate 27/32).

Scan test:

- use the information that rudeness-related parameters are sparse;
- false alarm rate 11/32.

	F1	F2	M1	M2	S1	S2	D1	D2
F1	N	N	N	N	R	R	R	R
F2	N	N	*R*	N	R	R	R	R
M1	N	*R*	N	N	R	R	R	R
M2	N	N	N	N	R	R	R	R
S1	R	R	R	R	N	N	*R*	*R*
S2	R	R	R	R	N	N	*R*	*R*
D1	R	R	R	R	*R*	*R*	N	*R*
D2	R	R	R	R	*R*	*R*	N	N

References I

George Box and José Ramírez.
Cumulative score charts.
Quality and Reliability Engineering International, 8(1):17–27, 1992.

David V Hinkley.
Inference about the change-point in a sequence of random variables.
Biometrika, 57(1):1–17, 1970.

Gary Lorden.
Procedures for reacting to a change in distribution.
The Annals of Mathematical Statistics, 42(6):1897–1908, 1971.

Ewan S Page.
Continuous inspection schemes.
Biometrika, 41(1/2):100–115, 1954.

Ewan S Page.
Estimating the point of change in a continuous process.
Biometrika, 44(2):248–252, 1957.

References II

Walter Andrew Shewhart.
Economic control of quality of manufactured product.
ASQ Quality Press, 1931.

Karl Stratos et al.
Model-based word embeddings from decompositions of count matrices.
In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1282–1291. The Association for Computer Linguistics, 2015.