# Choosing an Optimizer for Econometric Estimation

## A comparison of Newton, score, quasi-Newton, trust-region and cubic-regularization methods

Apoorva Lal Krabbs

2026-11-03

# 1 Introduction

An econometric estimator is often defined by the criterion it minimizes. Computing it also requires a choice about curvature, step length and stopping. Those choices can determine whether a program reaches a useful solution, stops early in a flat direction, or fails after evaluating an implausibly large trial point. A method that works well for one likelihood can behave quite differently after changing the units of a regressor or the starting value.

This experiment compares these choices on seven familiar estimation problems: OLS, ridge, binary logit, Poisson, linear IV/GMM, nonlinear least squares and synthetic control. The test set contains **84 simulated instances**, varying sample size, numerical stress and sample seed. General-purpose solvers share criteria, starting values and first-derivative callbacks; second-order methods additionally receive exact Hessians. Existing CrabbyMetrics estimator APIs provide a separate comparison of complete fitting routines. Every returned vector is checked independently of its solver's own convergence message.

The strongest result is about **reliability under stress**. GALAHAD's trust-region and cubic-regularization methods solve every smooth instance to the stated objective and gradient tolerances. The reference BFGS implementation does so too. The fastest accepted method varies by problem: safeguarded Newton frequently wins, and ensmallen L-BFGS is competitive despite receiving no Hessian. BHHH's good performance on logit and Poisson does not extend to distant starts for nonlinear least squares. Even among accepted fits, ill-conditioned OLS exposes an important distinction between accurately fitting the objective and precisely recovering its coefficients.

The sections below explain the algorithms, describe the experiment, and interpret the results. The supplement contains every family-level table, all convergence and recovery figures, and the numerical implementation.

# 2 What the optimizers do

Let $f(\beta)$ be the criterion to minimize, $g = \nabla f(\beta)$ its gradient, and $H = \nabla^2 f(\beta)$ its Hessian. Near the current point, the common starting point is the quadratic approximation

$$q(s) = f(\beta) + g's + \frac{1}{2}s'Hs.$$

The methods differ in how they represent curvature and how they limit the damage from an inaccurate local approximation.

## 2.1 Newton: use the observed curvature

Newton solves $Hs = -g$. For a positive-definite Hessian, this is the minimizer of the quadratic model. Near a well-behaved optimum it can converge rapidly; for a positive-definite quadratic, the exact unmodified step reaches the optimum in one iteration. An indefinite Hessian can instead produce an ascent direction, and a good local direction can still be too long for the nonlinear criterion.

The tested reference implementation is **safeguarded Newton**. It replaces Hessian eigenvalues below $10^{-5}$ by that floor and uses a line search to choose how far to move. The floor supplies a descent direction, but changes the step along flat directions. Its effects matter for the ill-conditioned OLS results; those results describe this implementation rather than every Newton solver.

## 2.2 BHHH: estimate curvature from individual scores

BHHH replaces the observed Hessian by an outer product of individual likelihood scores. For a mean negative log-likelihood, its curvature matrix is $C = n^{-1} \sum_i s_i s_{i'}$. The information identity connects this matrix to expected likelihood curvature at the correctly specified truth. They need not match in a finite sample or far from the optimum.

This avoids differentiating the score, at the cost of forming a score cross-product at each step. The reference BHHH implementation uses the same eigenvalue floor and line search as safeguarded Newton. Gaussian OLS and NLS require the variance rescaling implied by their squared-error normalization; ridge adds the penalty's curvature separately. BHHH is used for these likelihood-based problems, logit and Poisson. Fixed-weight IV/GMM receives no BHHH label.

## 2.3 DFP and BFGS: learn curvature from successive gradients

Quasi-Newton methods build an inverse-curvature approximation $M$ and use $d = -Mg$. After moving by $s$, they observe the gradient change $y$ and update $M$ to satisfy the inverse secant equation $M^+ y = s$.

**DFP** updates the inverse approximation directly:

$$M^+ = M + \frac{ss'}{s'y} - \frac{Myy'M}{y'My}.$$

**BFGS** uses a different update with the same secant target:

$$M^+ = (I - \rho sy')M(I - \rho ys') + \rho ss', \qquad \rho = (s'y)^{-1}.$$

Both retain positive definiteness when the curvature condition $s'y > 0$ holds. The tested implementations start at $M = I$, skip numerically degenerate updates, and reset a non-descent direction. Their line search is consequential: changing its curvature requirement materially changes DFP's success rate in this experiment.

## 2.4 L-BFGS: keep a short curvature history

L-BFGS uses a limited history of step/gradient-change pairs to apply an approximate inverse Hessian without storing a dense matrix. The tested implementations retain seven pairs. This is attractive when parameters are numerous or Hessians are expensive. Here it also provides a useful first-derivative-only comparator at modest dimension. ensmallen and CrabbyMetrics supply separate implementations with their own line-search and stopping rules.

## 2.5 TRU: minimize the quadratic inside a trust region

GALAHAD TRU solves a local step problem of the form

$$\min_{\|s\| \le \Delta} g's + \frac{1}{2} s'Hs.$$

The radius $\Delta$ limits the region over which the model is trusted. The algorithm compares the actual reduction in $f$ with the model's predicted reduction, accepting useful steps and adjusting the radius. The constraint also makes the step problem well posed when the Hessian is indefinite.

At a global solution of the Euclidean subproblem, the step obeys $(H + \lambda I)s = -g$ with $H + \lambda I \succcurlyeq 0$. Thus the method can change the direction through a curvature shift, as well as limiting its length. The benchmark uses exact dense Hessians and direct subproblem solves.

## 2.6 ARC: penalize long steps with a cubic term

GALAHAD ARC solves a regularized model of the form

$$\min_s g's + \frac{1}{2} s'Hs + \frac{\sigma}{3} \| s \|^3.$$

The cubic term discourages large steps and dominates negative quadratic curvature at large distances. Agreement between the model and the evaluated criterion determines how strongly subsequent steps are regularized. Its step also satisfies a shifted system, with $\lambda = \sigma \| s \|$ in Euclidean coordinates. TRU and ARC therefore share an adaptive curvature-shifting mechanism, while controlling the step through a radius and a penalty, respectively.

## 2.7 Step acceptance in the reference methods

Reference Newton, BHHH, DFP and BFGS share a strong-Wolfe line search. It asks for sufficient decrease in the objective and a reduction in the directional slope. The experiment compares the usual curvature parameter $c_2 = .9$ with the stricter $c_2 = .1$. The latter requires a slope closer to zero along the search direction; it can require more trial evaluations and does

not simply mean taking a shorter step. Package solvers retain their own controls and are timed once per repetition, without artificial duplication across the two reference settings.

### 2.8 The constrained synthetic-control methods

Synthetic control minimizes a quadratic over nonnegative donor weights that sum to one. The comparison therefore uses three constrained methods:

- **GALAHAD QPB** applies a primal–dual interior-point method to the quadratic program.
- **CrabbyMetrics' native solver** uses an active-set method, identifying weights on the boundary and solving for the remaining weights.
- **ensmallen Frank–Wolfe** moves toward a simplex vertex chosen from the current gradient, using convex combinations to preserve feasibility.

These methods exploit the actual constraint set. The unconstrained smooth solvers are outside this track. CrabbyMetrics' other native estimators likewise use their existing fitting routines and initializers; their timing covers a complete estimator API rather than isolating a generic optimizer.

## 3 Experimental design

Each family has two problem sizes, three stress levels and two sample seeds: twelve instances per family. The smooth problems use 2,000 or 20,000 observations and 8 or 24 coefficients, including an intercept. Synthetic control uses 100 or 500 periods and 12 or 50 donors.

| Estimator | Criterion | Stress control |
|---|---|---|
| OLS | Mean squared error | Design singular-value ratio: 1, 100, 10,000 |
| Ridge | Squared error plus slope penalty | Same samples as OLS; penalty 0.01 |
| Logit | Bernoulli negative log-likelihood | Slope norm: 1, 4, 12 |
| Poisson | Poisson negative log-likelihood | Regressor-unit ratio: 1, 100, 10,000 |
| IV/GMM | Fixed-weight quadratic moment criterion | First-stage strength: 1, 0.1, 0.01 |
| Nonlinear least squares | Exponential mean, additive Gaussian error | Starts: zero, $4\beta_0$, $-6\beta_0$ |
| Synthetic control | Squared error on the simplex | Donor noise SD: 1, 0.1, 0.01 |

**Table 1.** Stress has a different meaning in each row. Poisson changes units while preserving responses and true means. NLS changes starts while preserving the sample. IV varies identification. Logit samples are checked to exclude complete or quasi-separation.

There are 1,008 applicable method/instance/setting combinations. Each has one diagnostic warm-up and three timed repetitions in shuffled order, producing **3,024 timed fits**. Runs use float64 and one CPU thread on an Apple M4. Timing includes criterion preparation, solver setup and derivative work, including exact Hessians where supplied. Diagnostic trace logging is disabled during timed fits. Native estimator summaries are extracted after timing their fitting calls.

Smooth methods have a 1,000-iteration limit; callback-based fits also have a ten-second budget. Frank–Wolfe has 20,000 iterations. A successful smooth fit must have relative objective excess at most $10^{-8}$ and regressor-RMS-scaled gradient at most $10^{-6}$, using an independent reference. Synthetic control must have simplex violation at most $10^{-8}$ and a Frank–Wolfe certificate at most $10^{-7}$. **An instance counts as passed only when all three timed repetitions meet the external check.** Solver convergence messages are recorded separately. A small objective/gradient error alone need not ensure precise coefficients in a flat direction.

## 4 Results

### 4.1 Reliability depends on the implementation and the problem

TRU, ARC and reference BFGS pass all 72 smooth instances. The remaining methods show why a single success-rate number requires context: safeguarded Newton's misses are concentrated in ill-conditioned OLS, while BHHH has difficulty with distant NLS starts. Stronger Wolfe conditions help DFP substantially.

| Method | Wolfe .9 | Wolfe .1 | Package controls |
|---|---|---|---|
| Safeguarded Newton | 69/72 | 71/72 | — |
| BHHH | 51/60 | 49/60 | — |
| DFP | 54/72 | 68/72 | — |
| BFGS (reference) | 72/72 | 72/72 | — |
| GALAHAD TRU | — | — | 72/72 |
| GALAHAD ARC | — | — | 72/72 |
| ensmallen L-BFGS | — | — | 64/72 |
| CrabbyMetrics BFGS | — | — | 49/72 |
| CrabbyMetrics L-BFGS | — | — | 48/72 |

**Table 2.** Passed instances / attempted instances. BHHH's denominator is the 60 likelihood-based instances; the other rows cover 72 smooth instances. Native estimator APIs are reported separately in the supplement.

BHHH passes every logit and Poisson instance under both settings, but only four of twelve NLS instances. In the latter model, its score cross-product contains squared residuals as well as derivatives of the exponential mean. Far from a good fit, this curvature approximation can differ greatly from the observed Hessian. Its information-identity motivation provides no uniform guarantee over these starting points.

For DFP, stricter Wolfe search raises acceptance from 54 to 68 of 72 instances. NLS improves from four to twelve passes; IV/GMM improves from ten to twelve. The reference BFGS implementation passes every instance under either setting, while the existing CrabbyMetrics BFGS call passes 49. The label "BFGS" leaves important choices about step acceptance, initialization and termination open. These are comparisons of the stated implementations and tolerances.

## 4.2 Speed is meaningful only after the accuracy check

Figure 1 compares each solver with the fastest accepted solver on the *same* instance. Failed fits receive an infinite time ratio, so an early failure cannot appear as a speed advantage. The height at ratio one is the fraction of wins; the eventual height is the fraction of all instances passed.

With reference Wolfe .9, safeguarded Newton is fastest on 34 of the 72 smooth instances and ensmallen L-BFGS on 25. With Wolfe .1, those counts are 41 and 19. TRU and ARC provide broader reliability than either of these configurations, with extra subproblem work. The dimensions here are small enough that forming and factoring a Hessian can repay its cost through fewer iterations; the same balance can change at much larger parameter dimension.

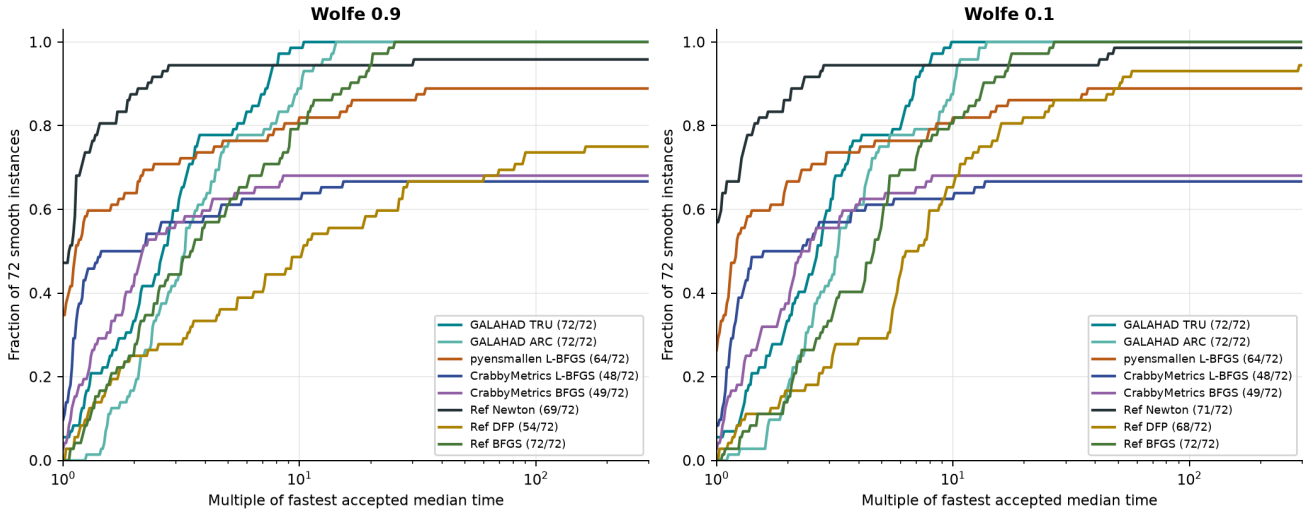



Figure 1: Matched runtime profiles on all 72 smooth instances. Each panel uses one reference Wolfe setting and the same package runs. Failures remain in the denominator. BHHH has a different 60-instance coverage set and is excluded from this profile; native estimator APIs also form a separate track.

## 4.3 Scaling and distant starts expose different weaknesses

In Poisson, changing regressor units leaves the statistical problem unchanged, yet ensmallen L-BFGS passes only eight of twelve instances and each generic CrabbyMetrics method passes four. The reference methods, TRU and ARC pass all twelve. This is a coordinate-scaling result rather than a deterioration in the data's information about the mean.

NLS instead holds the observations fixed and changes initialization. TRU, ARC, safeguarded Newton and reference BFGS pass all twelve instances. ensmallen L-BFGS passes eleven; the two generic CrabbyMetrics methods pass four each. Figure 2 shows a difficult starting point alongside the badly scaled Poisson case. The full timing panels below retain DFP, both Wolfe settings and all package paths, including failures.

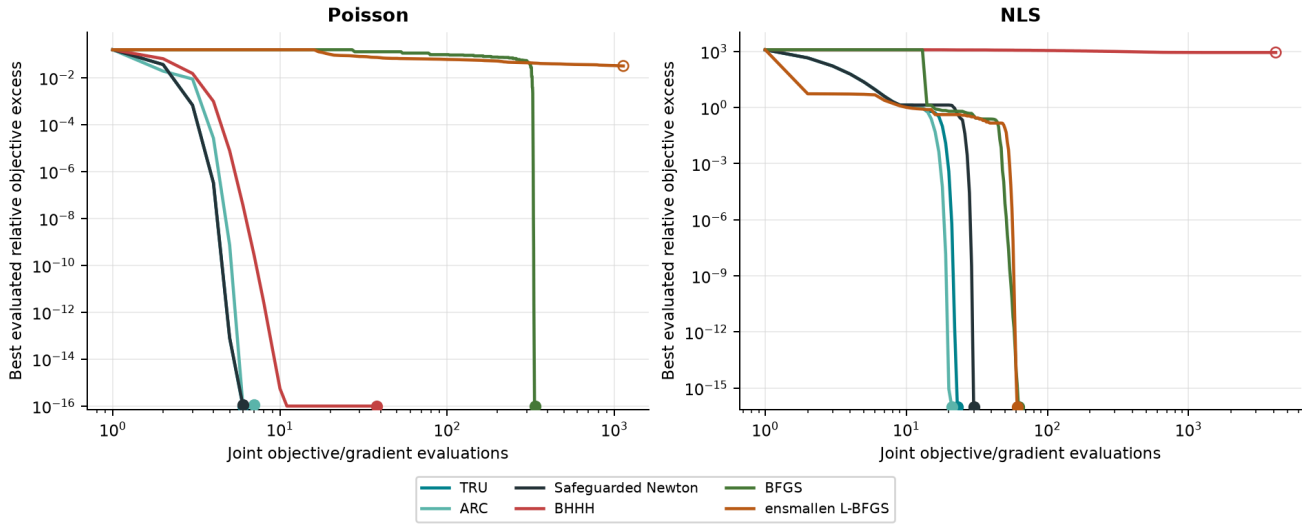



Figure 2: Selected methods on the larger stress-2, seed-0 Poisson and NLS instances (20,000 observations, 24 coefficients). References use Wolfe .9. Lines show the best evaluated objective, including trial points; circles use the returned vector, with open markers indicating rejection. Log-display excess is floored at 1e-16. Evaluation counts exclude the different amounts of curvature work; Figure 1 includes that work in timing.

## 4.4 Timing and convergence by estimator

The following panels report every estimator family's timing and convergence paths. Each uses the larger, stress-2, seed-0 instance: 20,000 observations and 24 coefficients for the smooth problems, or 500 periods and 50 donors for synthetic control. The 84-case atlas contains every size, stress and seed.

Within each figure, the left column measures joint objective/gradient evaluations and the right column measures elapsed time during the instrumented diagnostic warm-up. Smooth-problem rows use reference Wolfe .9 and .1; the package runs are shared across rows. Timing tables and Figure 1 use the separate, uninstrumented repetitions. Lines include trial points and trace the best evaluated objective. Circles rescore the returned vector; open markers fail the common accuracy test. QPB and native estimator APIs expose no callback path, so their time panels show returned endpoints only.

### 4.4.a OLS

Collinearity creates flat directions; a small objective error can coexist with slow coefficient adjustment.

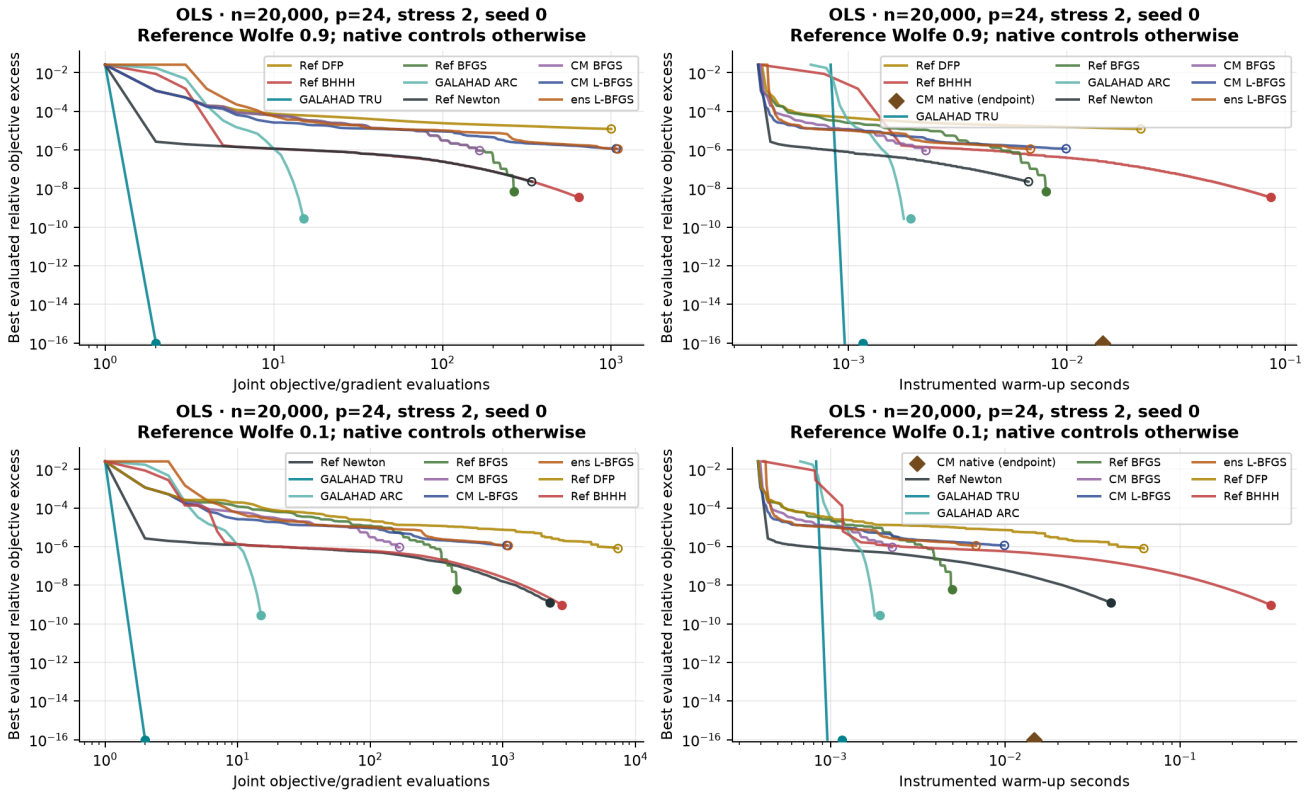



Figure 3: OLS: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.b Ridge

The slope penalty improves curvature on the same samples used for OLS.

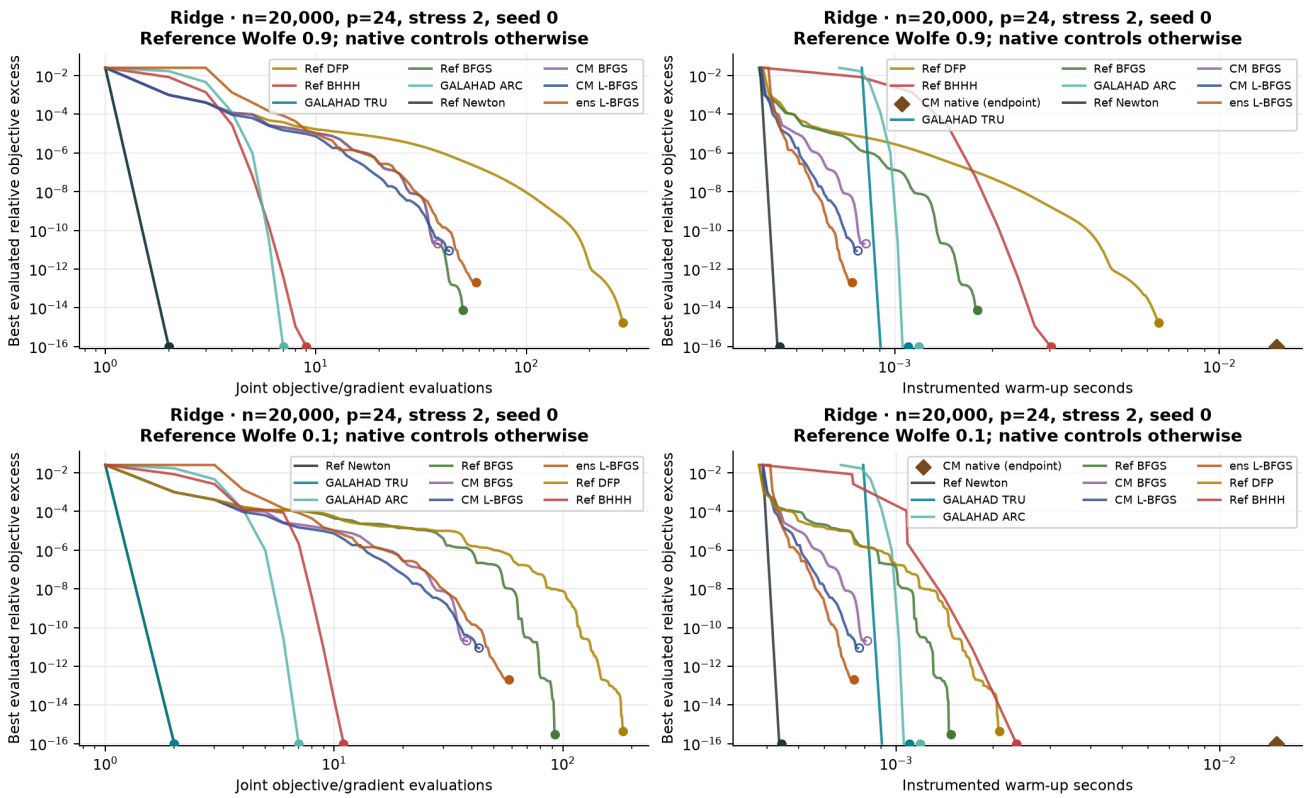



Figure 4: Ridge: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.c Logit

The strong-signal design weakens likelihood curvature while retaining a finite MLE.

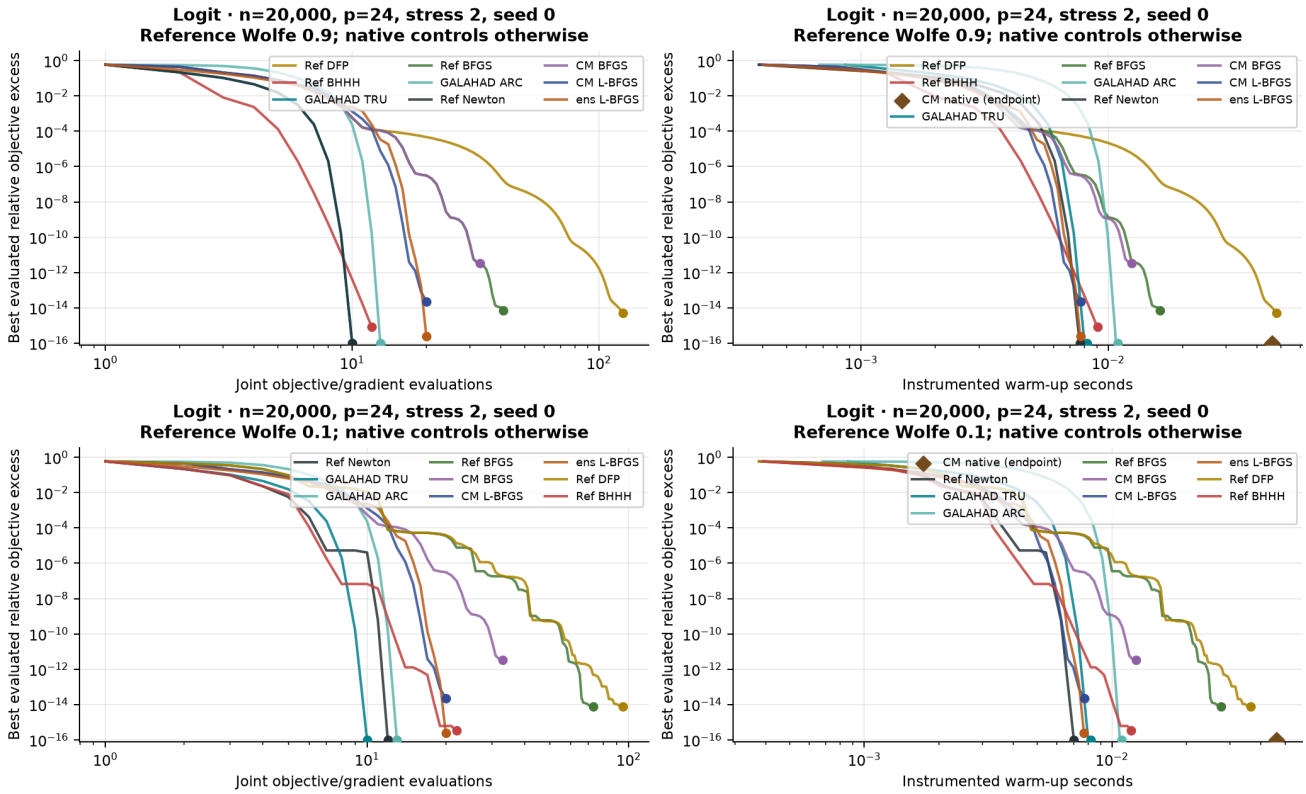



Figure 5: Logit: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.d Poisson

The highest-stress case changes regressor units while preserving the true mean function.

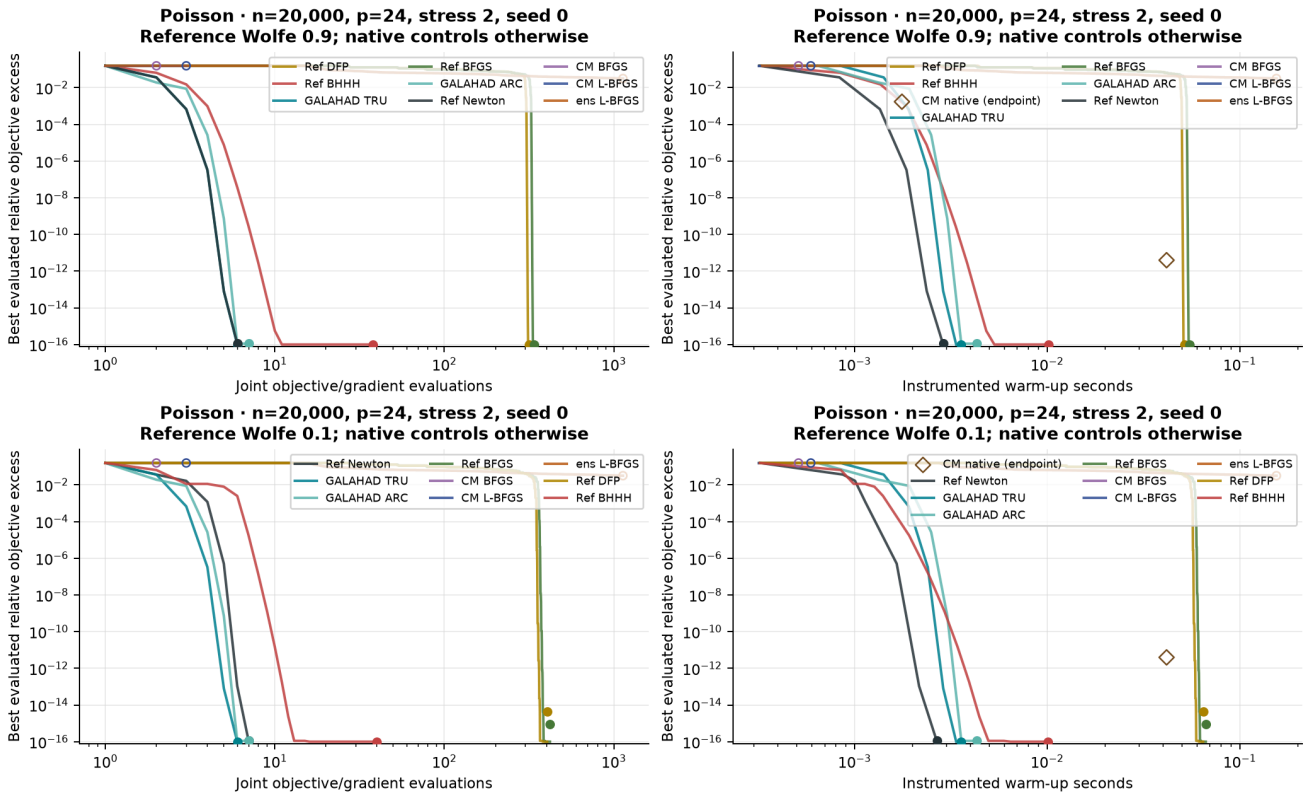



Figure 6: Poisson: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.e IV/GMM

The weak-first-stage case combines a quadratic sample criterion with limited statistical identification.

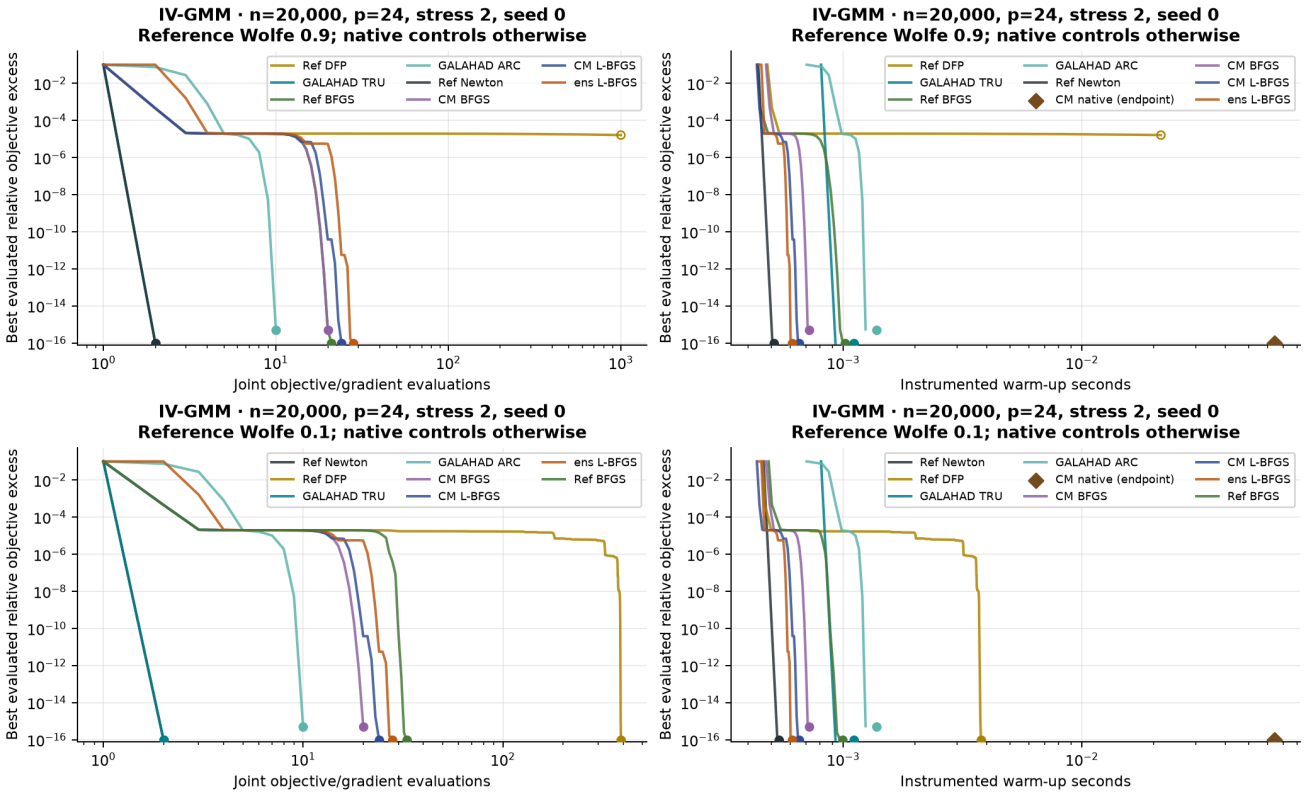



Figure 7: IV/GMM: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.f Nonlinear least squares

The distant initial coefficients test step control in the exponential-mean criterion.

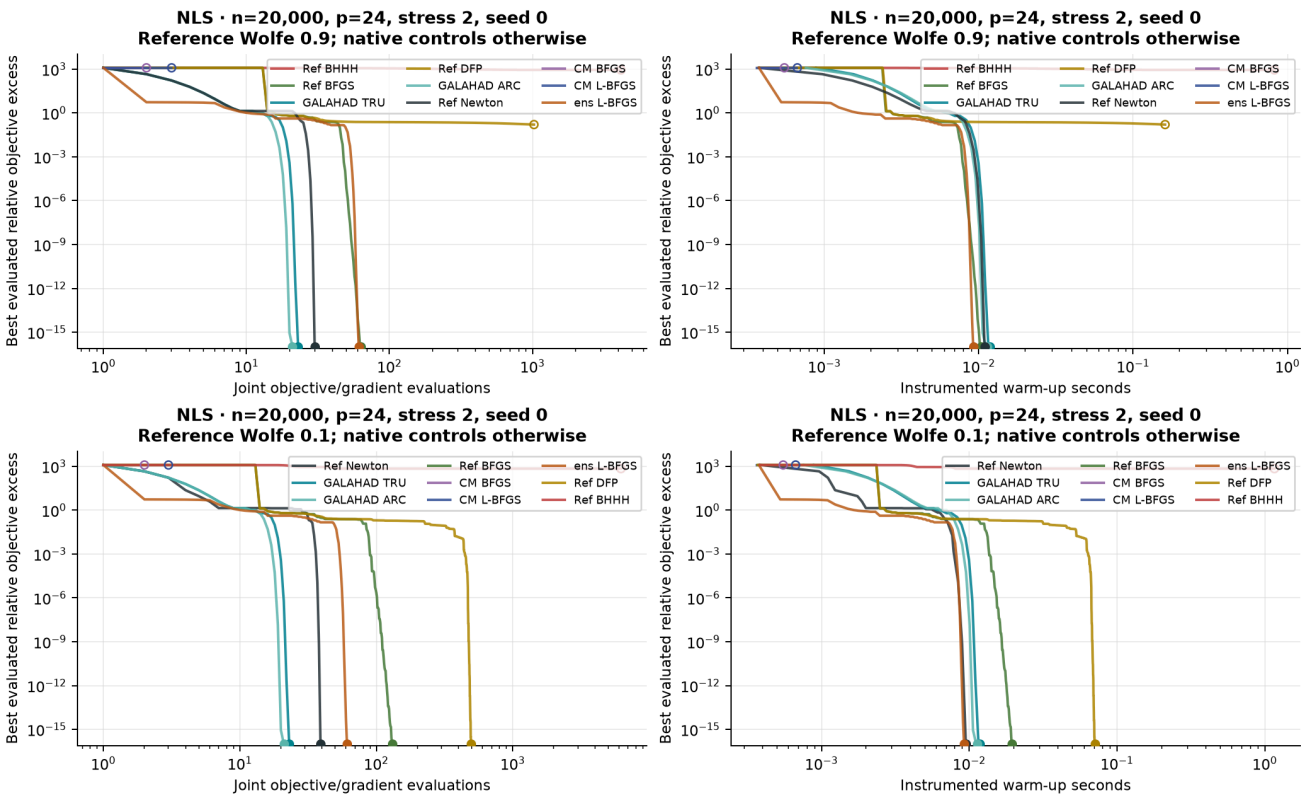



Figure 8: Nonlinear least squares: relative objective excess against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

### 4.4.g Synthetic control

All three methods enforce the simplex; the Frank–Wolfe gap supplies the convergence certificate.

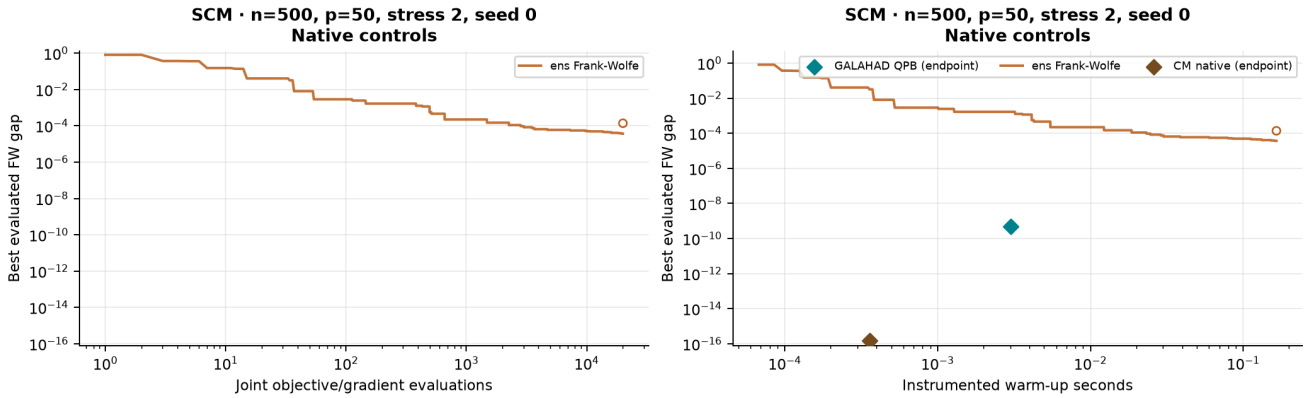



Figure 9: Synthetic control: Frank–Wolfe gap against evaluation count (left) and instrumented diagnostic elapsed time (right), with all applicable solvers. Returned endpoints are filled when accepted and open when rejected.

## 4.5 Solving the criterion does not remove identification problems

OLS and ridge clarify the distinction between numerical and statistical accuracy. All twelve ridge instances pass for the reference methods and both GALAHAD methods. Without the ridge penalty, collinearity makes a small gradient compatible with appreciable coefficient disagreement. Among accepted OLS fits, reference BFGS differs from the SVD coefficients by as much as 13.8% in relative Euclidean norm, and ARC by 8.8%. Their largest relative prediction disagreements are only about $1.2 \times 10^{-4}$ and $9.3 \times 10^{-5}$, respectively. The acceptance target controls objectives and gradients; applications requiring particular coefficients may need a stronger target.

Sampling error is a separate issue. Across the four IV instances at each strength level, the sample-reference estimate's median raw coefficient RMSE rises from 0.010 to 0.213 as instruments weaken. An optimizer can closely reproduce this sample estimate while remaining far from the generating coefficient. OLS collinearity produces an even sharper coefficient/prediction contrast. Figure 10 separates recovery against truth from numerical acceptance. The full seven-family recovery plots are in the supplement.

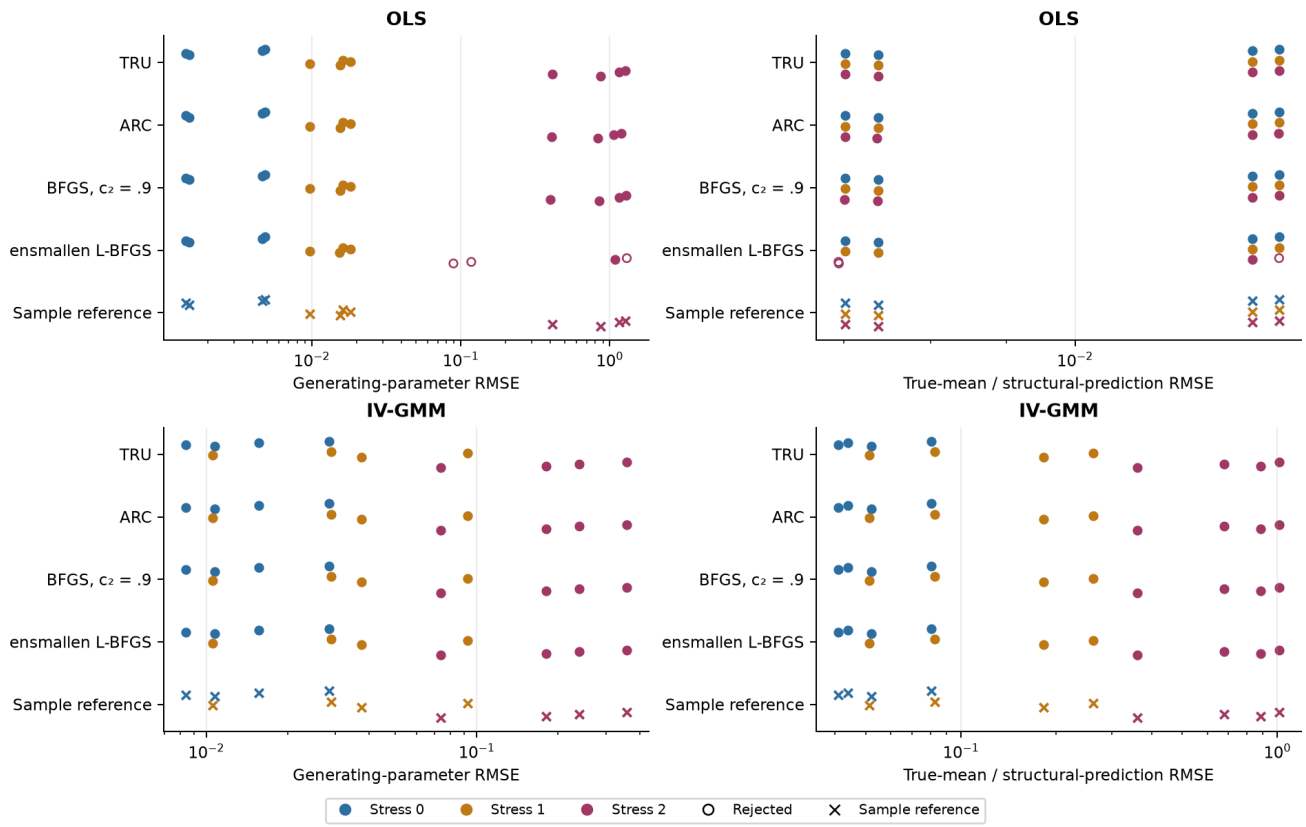



Figure 10: Recovery on all twelve OLS and IV/GMM instances for four representative methods. Each point is one method/setting and size/stress/seed, using the median metric over three timed repeats; open points fail at least one repeat. Colors identify stress and vertical offsets distinguish size/seed. Parameter RMSE averages coefficient errors in regressor-RMS units; prediction RMSE averages over observations. Reference crosses show the sample optimum's statistical error.

### 4.6 Synthetic control rewards specialized constrained solvers

Both QPB and CrabbyMetrics' native active-set solver satisfy the tight simplex certificate in all twelve instances. The native solver is faster in these small dense problems. Classic Frank–Wolfe misses the required gap within its fixed budget, despite preserving feasibility. A method's simplicity and low per-step cost therefore need to be assessed alongside the accuracy it actually reaches.

| Method | Passed | Median attempted time (ms) |
|---|---|---|
| GALAHAD QPB | 12/12 | 2.179 |
| CrabbyMetrics active set | 12/12 | 0.176 |
| ensmallen Frank–Wolfe | 0/12 | 93.493 |

**Table 3.** Median time across the twelve instances, after taking the median of three repetitions within each instance. Failed attempts are included in the time column; Frank–Wolfe's time is therefore a cost to reach its stopping point, without achieving the required solution accuracy.

As donor series become more alike, the reference weight RMSE rises while its prediction RMSE stays small. Several weight vectors can describe nearly the same fitted path. This is the constrained analogue of the flat-direction problem in OLS, and is visible in the full recovery diagnostics.

## 5 Implications and limits

For these small-to-moderate dense problems, TRU and ARC are reliable ways to use exact second derivatives across adverse units and starting points. Reference BFGS provides the same pass rate using first derivatives, with different timing and coefficient-accuracy behavior. Safeguarded Newton is often the fastest accepted method, while its fixed curvature floor merits attention in very flat problems. BHHH is attractive for some likelihoods, with performance that depends on the distance from a good fit. DFP illustrates how much a line-search choice can matter within a single update formula.

A practical choice should specify the software implementation, derivative information, scaling, starting values and accuracy target together. The existing CrabbyMetrics wrappers and native estimators are evaluated as supplied; the experiment makes no changes to their development code. Native estimator calls include their own preparation and fitting work, so their times do not isolate an optimizer or programming language.

These conclusions concern the sampled configurations: two seeds, modest dense parameter dimensions, one CPU thread and analytic derivatives. Large sparse models, matrix-free Hessian products, stochastic methods, GPUs and statistical inference are outside the comparison. The NLS reference is the best of three checked least-squares fits and carries no proof of global optimality. Three timing repeats are also a small sample, particularly for sub-millisecond fits.

## 6 Technical details and reproduction

The technical supplement contains each family's package and reference tables, complete convergence/recovery figures, numerical checks and expandable implementation code. The 84-panel atlas covers every size, stress and seed rather than only the representative cases shown here.

For a smooth criterion, the two acceptance conditions are

$$\frac{f\big(\hat\beta\big) - f\big(\hat\beta_{\mathrm{ref}}\big)}{\max\big\{1, \big|f\big(\hat\beta_{\mathrm{ref}}\big)\big|\big\}} \le 10^{-8}, \qquad \max_j \frac{\big|\nabla_j f\big(\hat\beta\big)\big|}{s_j} \le 10^{-6}, \quad s_j = \sqrt{n^{-1}\sum_i X_{ij}^2}.$$

For a feasible simplex point, the certificate is $\nabla f(w)'w - \min_j \nabla_j f(w) \le 10^{-7}$; it upper-bounds the convex objective's suboptimality. Feasibility is checked separately. Gaussian BHHH curvature is rescaled as $n^{-1}\sum_i r_i^2 a_i a_{i'}/v + P$, where $a_i$ is the derivative of the mean, $v$ the known simulated noise variance and $P$ the ridge-penalty Hessian. The noise SD is 0.25 for OLS/ridge and 0.1 for NLS.

OLS, ridge and IV references use augmented SVD least squares; logit and Poisson use an independent SciPy trust-exact solve in standardized coordinates. All 4,032 timed and diagnostic vectors are independently rescored. NLS uses a three-start residual-based least-squares reference, and synthetic control a constrained SLSQP reference with a simplex-gap check.

The Quarto source contains every generator, solver, check, table and plot; it reads no saved fit results or local helper modules. The replication bundle includes the executed notebook, pinned environment, timed fits, paired results, recovery metrics, evaluation traces, fitted vectors, validation and SHA-256 manifest.

Install Quarto and uv, then run with the bundled source and lockfile:

```
uv sync --locked
QUARTO_PYTHON="$PWD/.venv/bin/python" quarto render notes.qmd --to html
```

# 7 References


- Berndt, Hall, Hall and Hausman (1974), “Estimation and Inference in Nonlinear Structural Models.”
- Nocedal and Wright (2006), *Numerical Optimization*, Chapters 4 and 6.
- Cartis, Gould and Toint (2011), “Adaptive Cubic Regularisation Methods for Unconstrained Optimization. Part I.”
- GALAHAD documentation, version 5.5.3.
- pyensmallen, version 0.4.5.
- CrabbyMetrics, version 0.9.1.