Human review packet: GJ18InformativeRatingSystems

Generated from byte-pinned source excerpts, the expanded PaperInterface specifications, and hash-bound raw-source-to-Spec screening records.

Paper: GJ18InformativeRatingSystems
Paper id: GJ18InformativeRatingSystems
Source version: Byte-pinned arXiv 1810.13028 TeX plus the finite-chain corrected model recorded 2026-07-24
Source record: audit/paper_statement_map.json
Rows: 3
Generated: 2026-09-06
Source URL: official source

How to use this packet

For each source claim, compare the exact source excerpts with the expanded PaperInterface specification. Mark up this PDF or fill the blank reviewer box in the TeX source; return the annotated file for reconciliation into the paper-local human-review log.

Open the interactive dashboard

The interactive dashboard is an optional alternative to using this PDF; it presents the same review surface rather than an additional review requirement. After forking and cloning (or pulling) AppliedModelingLib, run the following from the repository root. The cache command is needed only the first time in a new clone.

```
cd <your-repository-checkout>
lake exe cache get
python3 scripts/review_dashboard.py --paper GJ18InformativeRatingSystems --serve
```

Open http://127.0.0.1:8765/ in a browser and keep that terminal running while reviewing. The dashboard records saved annotations in this paper's local reviewer-owned review record.

Contents

Governing Lean declarations

These exact graph-bound declaration bodies are retained by the displayed specifications or reviewed prerequisites. They are supporting context and do not add source-result rows or semantic judgments.

AppliedModelingLib.Probability.FiniteRatingLDPModel

Declaration location: reusable library

Lean declaration body

```
field typeLaw:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → θ → PMF Rating

field score:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → Rating → ℝ
```

AppliedModelingLib.Probability.finiteIidScoreSum

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → {ι : Type u_2} → [Fintype ι] → (α → ℝ) → (ι → α) → ℝ

value:
fun {α : Type u_1} {ι : Type u_2} [Fintype ι] (score : α → ℝ) (sample : ι → α) => ∑ i : ι, score (sample i)
```

AppliedModelingLib.Probability.finiteMGF

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (score : α → ℝ) (z : ℝ) => ∑ a : α, (μ a).toReal * Real.exp (z * score a)
```

AppliedModelingLib.Probability.finiteLogMGF

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (score : α → ℝ) (z : ℝ) =>
  Real.log (AppliedModelingLib.Probability.finiteMGF μ score z)
```

Direct retained dependencies

- AppliedModelingLib.Probability.finiteMGF

AppliedModelingLib.Probability.FiniteRatingLDPModel.logMGF

Declaration location: reusable library

Lean declaration body

```
type:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → θ → ℝ → ℝ

value:
fun {θ : Type u_2} {Rating : Type u_3} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating) (t : θ) (z : ℝ) =>
  AppliedModelingLib.Probability.finiteLogMGF (M.typeLaw t) M.score z
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteLogMGF

AppliedModelingLib.Probability.finiteLegendreValue

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ → ℝ → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (score : α → ℝ) (a z : ℝ) =>
  z * a - AppliedModelingLib.Probability.finiteLogMGF μ score z
```

Direct retained dependencies

- AppliedModelingLib.Probability.finiteLogMGF

AppliedModelingLib.Probability.finiteRateFunction

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (score : α → ℝ) (a : ℝ) =>
  sSup (Set.range fun (z : ℝ) => AppliedModelingLib.Probability.finiteLegendreValue μ score a z)
```

Direct retained dependencies

- AppliedModelingLib.Probability.finiteLegendreValue

AppliedModelingLib.Probability.FiniteRatingLDPModel.rateFunction

Declaration location: reusable library

Lean declaration body

```
type:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → θ → ℝ → ℝ

value:
fun {θ : Type u_2} {Rating : Type u_3} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating) (t : θ) (a : ℝ) =>
  AppliedModelingLib.Probability.finiteRateFunction (M.typeLaw t) M.score a
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteRateFunction

AppliedModelingLib.Probability.finiteRateFunctionTop

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ → WithTop ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (score : α → ℝ) (a : ℝ) =>
  sSup (Set.range fun (z : ℝ) => ↑(AppliedModelingLib.Probability.finiteLegendreValue μ score a z))
```

Direct retained dependencies

- AppliedModelingLib.Probability.finiteLegendreValue

AppliedModelingLib.Probability.FiniteRatingLDPModel.rateFunctionTop

Declaration location: reusable library

Lean declaration body

```
type:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → θ → ℝ → WithTop ℝ

value:
fun {θ : Type u_2} {Rating : Type u_3} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating) (t : θ) (a : ℝ) =>
  AppliedModelingLib.Probability.finiteRateFunctionTop (M.typeLaw t) M.score a
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteRateFunctionTop

AppliedModelingLib.Probability.finiteChainJointFloorRatingSample

Declaration location: reusable library

Lean declaration body

```
type:
{n : ℕ} → Type u_1 → (Fin n → ℝ) → ℕ → Type u_1

value:
fun {n : ℕ} (Rating : Type u_1) (sampleRate : Fin n → ℝ) (k : ℕ) =>
  (θ : Fin n) → Fin (AppliedModelingLib.Probability.floorSampleCount sampleRate θ k) → Rating
```

Direct retained dependencies

- AppliedModelingLib.Probability.floorSampleCount

AppliedModelingLib.Probability.logDecay

Declaration location: reusable library

Lean declaration body

```
type:
(ℕ → ℝ) → ℕ → ℝ

value:
fun (p : ℕ → ℝ) (n : ℕ) => -(↑n)⁻¹ * Real.log (p n)
```

AppliedModelingLib.Probability.HasExponentialRate

Declaration location: reusable library

Lean declaration body

```
type:
(ℕ → ℝ) → ℝ → Prop

value:
fun (p : ℕ → ℝ) (rate : ℝ) => Filter.Tendsto (AppliedModelingLib.Probability.logDecay p) Filter.atTop (nhds rate)
```

Direct retained dependencies

- AppliedModelingLib.Probability.logDecay

AppliedModelingLib.Probability.ExponentialRateCertificate

Declaration location: reusable library

Lean declaration body

```
field eventually_pos:
∀ {p : ℕ → ℝ} {rate : ℝ},
  AppliedModelingLib.Probability.ExponentialRateCertificate p rate → ∀ᶠ (n : ℕ) in Filter.atTop, 0 < p n

field has_rate:
∀ {p : ℕ → ℝ} {rate : ℝ},
  AppliedModelingLib.Probability.ExponentialRateCertificate p rate →
    AppliedModelingLib.Probability.HasExponentialRate p rate
```

Direct retained dependencies

- AppliedModelingLib.Probability.HasExponentialRate

AppliedModelingLib.Probability.withTopRealScale

Declaration location: reusable library

Lean declaration body

```
type:
ℝ → WithTop ℝ → WithTop ℝ

value:
fun (c : ℝ) (x : WithTop ℝ) => WithTop.map (fun (r : ℝ) => c * r) x
```

AppliedModelingLib.Probability.FiniteRatingLDPModel.pairwiseRateObjectiveTop

Declaration location: reusable library

Lean declaration body

```
type:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → (θ → ℝ) → θ → θ → ℝ → WithTop ℝ

value:
fun {θ : Type u_2} {Rating : Type u_3} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating) (sampleRate : θ → ℝ) (hi lo : θ) (a : ℝ) =>
  AppliedModelingLib.Probability.withTopRealScale (sampleRate hi) (M.rateFunctionTop hi a) +
    AppliedModelingLib.Probability.withTopRealScale (sampleRate lo) (M.rateFunctionTop lo a)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.rateFunctionTop
- AppliedModelingLib.Probability.withTopRealScale

AppliedModelingLib.Probability.FiniteRatingLDPModel.pairwiseThresholdRateTop

Declaration location: reusable library

Lean declaration body

```
type:
{θ : Type u_2} →
  {Rating : Type u_3} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating → (θ → ℝ) → θ → θ → WithTop ℝ

value:
fun {θ : Type u_2} {Rating : Type u_3} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel θ Rating) (sampleRate : θ → ℝ) (hi lo : θ) =>
  sInf (Set.range fun (a : ℝ) => M.pairwiseRateObjectiveTop sampleRate hi lo a)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.pairwiseRateObjectiveTop

AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop

Declaration location: reusable library

Lean declaration body

```
type:
{Seller : Type u_1} →
  {Rating : Type u_2} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating → (Seller → ℝ) → Seller → Seller → WithTop ℝ

value:
fun {Seller : Type u_1} {Rating : Type u_2} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating) (sampleRate : Seller → ℝ)
    (hi lo : Seller) =>
  M.pairwiseThresholdRateTop sampleRate hi lo
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.pairwiseThresholdRateTop

AppliedModelingLib.pmfExp

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (α → ℝ) → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (f : α → ℝ) => ∑ a : α, (μ a).toReal * f a
```

AppliedModelingLib.pmfPi

Declaration location: reusable library

Lean declaration body

```
type:
{ι : Type u_1} →
  [Fintype ι] →
    [DecidableEq ι] → {α : ι → Type u_2} → [(i : ι) → Fintype (α i)] → ((i : ι) → PMF (α i)) → PMF ((i : ι) → α i)

value:
fun {ι : Type u_1} [Fintype ι] [DecidableEq ι] {α : ι → Type u_2} [(i : ι) → Fintype (α i)] (μ : (i : ι) → PMF (α i)) =>
  PMF.ofFintype (fun (f : (i : ι) → α i) => ∏ i : ι, (μ i) (f i)) (AppliedModelingLib.pmfPi._proof_1 μ)
```

AppliedModelingLib.pmfProb

Declaration location: reusable library

Lean declaration body

```
type:
{α : Type u_1} → [Fintype α] → PMF α → (p : α → Prop) → [DecidablePred p] → ℝ

value:
fun {α : Type u_1} [Fintype α] (μ : PMF α) (p : α → Prop) [DecidablePred p] =>
  AppliedModelingLib.pmfExp μ fun (a : α) => if p a then 1 else 0
```

Direct retained dependencies

- AppliedModelingLib.pmfExp

AppliedModelingLib.pmfProduct

Declaration location: reusable library

Lean declaration body

```
type:
(ι : Type u_1) → (α : Type u_2) → [Fintype ι] → [DecidableEq ι] → [Fintype α] → PMF α → PMF (ι → α)

value:
fun (ι : Type u_1) (α : Type u_2) [Fintype ι] [DecidableEq ι] [Fintype α] (μ : PMF α) =>
  PMF.ofFintype (fun (f : ι → α) => ∏ i : ι, μ (f i)) (AppliedModelingLib.pmfProduct._proof_1 ι α μ)
```

AppliedModelingLib.Probability.finiteChainJointFloorRatingLaw

Declaration location: reusable library

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (sampleRate : Fin n → ℝ) →
            (k : ℕ) → PMF (AppliedModelingLib.Probability.finiteChainJointFloorRatingSample Rating sampleRate k)

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ) (k : ℕ) =>
  AppliedModelingLib.pmfPi fun (θ : Fin n) =>
    AppliedModelingLib.pmfProduct (Fin (AppliedModelingLib.Probability.floorSampleCount sampleRate θ k)) Rating
      (M.typeLaw θ)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteChainJointFloorRatingSample
- AppliedModelingLib.Probability.floorSampleCount
- AppliedModelingLib.pmfPi
- AppliedModelingLib.pmfProduct

AppliedModelingLib.uniformPMF

Declaration location: reusable library

Lean declaration body

```
type:
(α : Type u_1) → [Fintype α] → [Nonempty α] → PMF α

value:
fun (α : Type u_1) [Fintype α] [Nonempty α] =>
  PMF.ofFintype (fun (x : α) => (↑(Fintype.card α))⁻¹) (AppliedModelingLib.uniformPMF._proof_1 α)
```

GJ18InformativeRatingSystems.clarifiedMatchRateBounds

Declaration location: paper-local

Lean declaration body

type:
{n : ℕ} → (Fin n → ℝ) → Prop

value:
fun {n : ℕ} (sampleRate : Fin n → ℝ) => ∀ (theta : Fin n), 0 < sampleRate theta ∧ sampleRate theta ≤ 1

GJ18InformativeRatingSystems.clarifiedNontrivialRatingScale

Declaration location: paper-local

Lean declaration body

type:
ℕ → Prop

value:
fun (m : ℕ) => 1 ≤ m

GJ18InformativeRatingSystems.clarifiedUniformTypePrior

Declaration location: paper-local

Lean declaration body

type:
(n : ℕ) → 0 < n → PMF (Fin n)

value:
fun (n : ℕ) (hn : 0 < n) => AppliedModelingLib.uniformPMF (Fin n)

Direct retained dependencies

- AppliedModelingLib.uniformPMF

GJ18InformativeRatingSystems.finiteChainAdjacentIndex

Declaration location: paper-local

Lean declaration body

type:
ℕ → Type

value:
fun (n : ℕ) => { i : Fin n // ↑i + 1 < n }

GJ18InformativeRatingSystems.finiteChainOrderedPair

Declaration location: paper-local

Lean declaration body

type:
ℕ → Type

value:
fun (n : ℕ) => { p : Fin n × Fin n // ↑p.2 < ↑p.1 }

GJ18InformativeRatingSystems.finiteChainAdjacentPair

Declaration location: paper-local

Lean declaration body

type:
{n : ℕ} →
  GJ18InformativeRatingSystems.finiteChainAdjacentIndex n → GJ18InformativeRatingSystems.finiteChainOrderedPair n

value:
fun {n : ℕ} (i : GJ18InformativeRatingSystems.finiteChainAdjacentIndex n) =>
  ((⟨↑↑i + 1, GJ18InformativeRatingSystems.finiteChainAdjacentPair._proof_1 i⟩, ↑i),
    GJ18InformativeRatingSystems.finiteChainAdjacentPair._proof_2 i⟩

Direct retained dependencies

- GJ18InformativeRatingSystems.finiteChainAdjacentIndex
- GJ18InformativeRatingSystems.finiteChainOrderedPair

GJ18InformativeRatingSystems.finiteChainOrderedPairHi

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} → GJ18InformativeRatingSystems.finiteChainOrderedPair n → Fin n

value:
fun {n : ℕ} (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) => (↑p).1
```

Direct retained dependencies

- GJ18InformativeRatingSystems.finiteChainOrderedPair

GJ18InformativeRatingSystems.finiteChainOrderedPairLo

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} → GJ18InformativeRatingSystems.finiteChainOrderedPair n → Fin n

value:
fun {n : ℕ} (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) => (↑p).2
```

Direct retained dependencies

- GJ18InformativeRatingSystems.finiteChainOrderedPair

GJ18InformativeRatingSystems.finiteChainOrderedPairThresholdRateTop

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (Fin n → ℝ) → GJ18InformativeRatingSystems.finiteChainOrderedPair n → WithTop ℝ

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ)
    (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) =>
  AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop M sampleRate
    (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p) (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo

GJ18InformativeRatingSystems.finiteChainAdjacentThresholdRateTop

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (Fin n → ℝ) → GJ18InformativeRatingSystems.finiteChainAdjacentIndex n → WithTop ℝ

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ)
    (i : GJ18InformativeRatingSystems.finiteChainAdjacentIndex n) =>
  GJ18InformativeRatingSystems.finiteChainOrderedPairThresholdRateTop M sampleRate
    (GJ18InformativeRatingSystems.finiteChainAdjacentPair i)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.finiteChainAdjacentIndex
- GJ18InformativeRatingSystems.finiteChainAdjacentPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairThresholdRateTop

GJ18InformativeRatingSystems.iidFloorSourceStateCarrier

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} → Type u_1 → (Fin n → ℝ) → ℕ → Type u_1

value:
fun {n : ℕ} (Rating : Type u_1) (sampleRate : Fin n → ℝ) (k : ℕ) =>
  AppliedModelingLib.Probability.finiteChainJointFloorRatingSample Rating sampleRate k
```

Direct retained dependencies

- AppliedModelingLib.Probability.finiteChainJointFloorRatingSample

GJ18InformativeRatingSystems.iidFloorSourceStateLaw

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (sampleRate : Fin n → ℝ) →
            (k : ℕ) → PMF (GJ18InformativeRatingSystems.iidFloorSourceStateCarrier Rating sampleRate k)

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ) (k : ℕ) =>
  AppliedModelingLib.Probability.finiteChainJointFloorRatingLaw M sampleRate k
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteChainJointFloorRatingLaw
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier

GJ18InformativeRatingSystems.iidFloorSourceStateScore

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (sampleRate : Fin n → ℝ) →
            (k : ℕ) → GJ18InformativeRatingSystems.iidFloorSourceStateCarrier Rating sampleRate k → Fin n → ℝ

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ) (k : ℕ)
    (a : GJ18InformativeRatingSystems.iidFloorSourceStateCarrier Rating sampleRate k) (a_1 : Fin n) =>
  AppliedModelingLib.Probability.finiteChainJointFloorAverageScore M sampleRate k a a_1
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteChainJointFloorAverageScore
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier

GJ18InformativeRatingSystems.minFiniteChainAdjacentThresholdRateTop

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        [Nonempty (GJ18InformativeRatingSystems.finiteChainAdjacentIndex n)] →
          AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating → (Fin n → ℝ) → WithTop ℝ

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    [Nonempty (GJ18InformativeRatingSystems.finiteChainAdjacentIndex n)]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ) =>
  Finset.univ.inf' GJ18InformativeRatingSystems.minFiniteChainAdjacentThresholdRate._proof_1
    (GJ18InformativeRatingSystems.finiteChainAdjacentThresholdRateTop M sampleRate)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.finiteChainAdjacentIndex
- GJ18InformativeRatingSystems.finiteChainAdjacentThresholdRateTop

GJ18InformativeRatingSystems.sourceOrdinalUpperTail

Declaration location: paper-local

Lean declaration body

```
type:
{Seller : Type u_1} →
  {m : ℕ} → AppliedModelingLib.Probability.FiniteRatingLDPModel Seller (Fin (m + 1)) → Seller → Fin (m + 1) → ℝ

value:
fun {Seller : Type u_1} {m : ℕ} (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller (Fin (m + 1)))
    (θ : Seller) (t : Fin (m + 1)) =>
  AppliedModelingLib.pmfProb (M.typeLaw θ) fun (y : Fin (m + 1)) => t ≤ y
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.pmfProb

GJ18InformativeRatingSystems.sourceStateScoreGapLeftTailProb

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} →
  {Omega : Type u_1} →
    [Fintype Omega] →
      [DecidableEq Omega] → PMF Omega → (Omega → Fin n → ℝ) → GJ18InformativeRatingSystems.finiteChainOrderedPair n → ℝ

value:
fun {n : ℕ} {Omega : Type u_1} [Fintype Omega] [DecidableEq Omega] (mu : PMF Omega) (score : Omega → Fin n → ℝ)
    (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) =>
  AppliedModelingLib.pmfProb mu fun (omega : Omega) =>
    score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p) ≤
      score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p)
```

Direct retained dependencies

- AppliedModelingLib.pmfProb
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo

GJ18InformativeRatingSystems.source_at_most_one_match_per_period

Declaration location: paper-local

Lean declaration body

```
type:
{Seller : Type u_1} → (Seller → ℝ) → Prop

value:
fun {Seller : Type u_1} (sampleRate : Seller → ℝ) =>
  ∀ (θ : Seller) (k : ℕ),
    AppliedModelingLib.Probability.floorSampleCount sampleRate θ (k + 1) =
        AppliedModelingLib.Probability.floorSampleCount sampleRate θ k ∨
      AppliedModelingLib.Probability.floorSampleCount sampleRate θ (k + 1) =
        AppliedModelingLib.Probability.floorSampleCount sampleRate θ k + 1
```

Direct retained dependencies

- AppliedModelingLib.Probability.floorSampleCount

GJ18InformativeRatingSystems.source_monotone_match_rates

Declaration location: paper-local

Lean declaration body

```
type:
{n : ℕ} → (Fin n → ℝ) → Prop

value:
fun {n : ℕ} (sampleRate : Fin n → ℝ) => Monotone sampleRate
```

GJ18InformativeRatingSystems.source_nonnegative_match_rates

Declaration location: paper-local

Lean declaration body

```
type:
{Seller : Type u_1} → (Seller → ℝ) → Prop

value:
fun {Seller : Type u_1} (sampleRate : Seller → ℝ) => ∀ (θ : Seller), 0 ≤ sampleRate θ
```

GJ18InformativeRatingSystems.source_match_schedule_coherence

Declaration location: paper-local

Lean declaration body

type:
```
{n : ℕ} → (Fin n → ℝ) → Prop
```

value:
```
fun {n : ℕ} (sampleRate : Fin n → ℝ) =>
  GJ18InformativeRatingSystems.source_nonnegative_match_rates sampleRate ∧
    GJ18InformativeRatingSystems.source_monotone_match_rates sampleRate ∧
      GJ18InformativeRatingSystems.source_at_most_one_match_per_period sampleRate
```

Direct retained dependencies

- GJ18InformativeRatingSystems.source_at_most_one_match_per_period
- GJ18InformativeRatingSystems.source_monotone_match_rates
- GJ18InformativeRatingSystems.source_nonnegative_match_rates

GJ18InformativeRatingSystems.source_rating_scores_in_unit_interval

Declaration location: paper-local

Lean declaration body

type:
```
{Rating : Type u_1} → (Rating → ℝ) → Prop
```

value:
```
fun {Rating : Type u_1} (score : Rating → ℝ) => ∀ (r : Rating), 0 ≤ score r ∧ score r ≤ 1
```

GJ18InformativeRatingSystems.source_strict_ordinal_cumulative_tail_decrease_at_displayed_cutoffs

Declaration location: paper-local

Lean declaration body

type:
```
{Seller : Type u_1} → {m : ℕ} → AppliedModelingLib.Probability.FiniteRatingLDPModel Seller (Fin (m + 1)) → Prop
```

value:
```
fun {Seller : Type u_1} {m : ℕ} (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller (Fin (m + 1))) =>
  ∀ (θ : Seller) (r : Fin m),
    GJ18InformativeRatingSystems.sourceOrdinalUpperTail M θ r.succ <
      GJ18InformativeRatingSystems.sourceOrdinalUpperTail M θ r.castSucc
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.sourceOrdinalUpperTail

GJ18InformativeRatingSystems.source_strict_ordinal_quality_tail_increase_above_bottom

Declaration location: paper-local

Lean declaration body

type:
```
{n m : ℕ} → AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1)) → Prop
```

value:
```
fun {n m : ℕ} (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))) =>
  ∀ (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) (t : Fin (m + 1)),
    0 < ↑t →
      GJ18InformativeRatingSystems.sourceOrdinalUpperTail M (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p)
          t <
        GJ18InformativeRatingSystems.sourceOrdinalUpperTail M (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p)
          t
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo
- GJ18InformativeRatingSystems.sourceOrdinalUpperTail

GJ18InformativeRatingSystems.source_strictly_increasing_rating_scores

Declaration location: paper-local

Lean declaration body

```
type:
{Rating : Type u_1} → [Preorder Rating] → (Rating → ℝ) → Prop

value:
fun {Rating : Type u_1} [Preorder Rating] (score : Rating → ℝ) => StrictMono score
```

GJ18InformativeRatingSystems.FiniteOrdinalSourceModel

Declaration location: paper-local

Lean declaration body

```
field match_schedule:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ},
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    GJ18InformativeRatingSystems.source_match_schedule_coherence sampleRate

field scores_in_unit_interval:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ},
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    GJ18InformativeRatingSystems.source_rating_scores_in_unit_interval M.score

field scores_strictly_increasing:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ},
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    GJ18InformativeRatingSystems.source_strictly_increasing_rating_scores M.score

field tails_strictly_decrease_at_displayed_cutoffs:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ},
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    GJ18InformativeRatingSystems.source_strict_ordinal_cumulative_tail_decrease_at_displayed_cutoffs M

field quality_tails_strictly_increase_above_bottom:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ},
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    GJ18InformativeRatingSystems.source_strict_ordinal_quality_tail_increase_above_bottom M
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.source_match_schedule_coherence
- GJ18InformativeRatingSystems.source_rating_scores_in_unit_interval
- GJ18InformativeRatingSystems.source_strict_ordinal_cumulative_tail_decrease_at_displayed_cutoffs
- GJ18InformativeRatingSystems.source_strict_ordinal_quality_tail_increase_above_bottom
- GJ18InformativeRatingSystems.source_strictly_increasing_rating_scores

GJ18InformativeRatingSystems.ClarifiedSourceModel.correctedTheorem1Rate

Declaration location: paper-local

Lean declaration body

```
type:
{n m : ℕ} →
  {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} →
    {sampleRate : Fin n → ℝ} → GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate → WithTop ℝ

value:
fun {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
    (model : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate) =>
  GJ18InformativeRatingSystems.minFiniteChainAdjacentThresholdRateTop M sampleRate
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.ClarifiedSourceModel
- GJ18InformativeRatingSystems.finiteChainAdjacentIndex
- GJ18InformativeRatingSystems.minFiniteChainAdjacentThresholdRateTop

Material library prerequisites

AppliedModelingLib.Probability.floorSampleCount

Source locator: cited publication, lines 669-670

Verbatim paper-source connection

```
{\bf Rating accumulation}.  Sellers accumulate ratings over time by matching with buyers. At each time step, each seller matches with at most a single
↪    buyer. We make one key assumption that drives the accumulation of ratings: in particular, that sellers of higher quality are more likely to be matched.
↪    %
We consider an analysis that is asymptotic in the number of ratings received by sellers and so we model this visibility benefit by assuming that {\em
↪    sellers of higher quality accumulate ratings at a faster rate}. In particular, we assume the existence of a nondecreasing {\em match function}
↪    $g(\theta)$, where a seller of type $\theta$ receives $n_k(\theta) = \lfloor k g(\theta) \rfloor$ matches, and thus ratings, up to time $k$.
```

Lean-expanded library semantic target

```
type:
{Seller : Type u_1} → (Seller → ℝ) → Seller → ℕ → ℕ

value:
fun {Seller : Type u_1} (sampleRate : Seller → ℝ) (θ : Seller) (k : ℕ) => ⌊↑k * sampleRate θ⌋₊
```

Exact Lean library declaration

```
/-- Paper sample count `n_k(theta) = floor(k * g(theta))`. -/
def floorSampleCount {Seller : Type*} (sampleRate : Seller → ℝ)
    (θ : Seller) (k : ℕ) : ℕ :=
  Nat.floor ((k : ℝ) * sampleRate θ)
```

Recorded source-to-library screening **Verdict:** matches

Reason: Matches the literal source formula n_k(theta)=floor(k g(theta)): the expanded definition is exactly the natural floor of k times the seller-specific match rate. The source's monotonicity and at-most-one-match discussion is not silently promoted into this formula declaration.

Reviewer check: ☐ Matches source input

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

AppliedModelingLib.Probability.finiteChainJointFloorAverageScore

Source locator: cited publication, lines 697-702

Verbatim paper-source connection

```
Let $y_0(\theta), y_1(\theta), y_2(\theta), \ldots$ be the sequence of ratings received by the seller of type $\theta$.  The {\em aggregate score} up to time
↪   $k$ of this seller is the average score from ratings received:
\begin{equation}
\label{eq:aggscore}
 x_k(\theta) =  \frac{1}{n_k(\theta)} \sum_{\ell = 0}^{n_k(\theta)} \phi(y_\ell(\theta)).
\end{equation}
(We presume $x_0(\theta) = 0$ for all $\theta$.)  Since $\phi(y) \in [0,1]$ for all $y$, the score $x_k$ also lies in $[0,1]$.
```

Formalized review target The archival source above is not asserted equivalent to this formalized target.

```
For each seller type theta and horizon k, if n_k(theta) = floor(k g(theta)), the aggregate score averages exactly n_k(theta) rating scores over a finite family
↪   indexed by Fin n_k(theta); the zero-sample convention remains explicit.
```

Archival source anchor: cited publication, lines 697-702

Lean-expanded library semantic target

```
type:
{n : ℕ} →
  {Rating : Type u_1} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating →
          (sampleRate : Fin n → ℝ) →
            (k : ℕ) → AppliedModelingLib.Probability.finiteChainJointFloorRatingSample Rating sampleRate k → Fin n → ℝ

value:
fun {n : ℕ} {Rating : Type u_1} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ) (k : ℕ)
    (sample : AppliedModelingLib.Probability.finiteChainJointFloorRatingSample Rating sampleRate k) (θ : Fin n) =>
  (↑(AppliedModelingLib.Probability.floorSampleCount sampleRate θ k))⁻¹ *
    AppliedModelingLib.Probability.finiteIidScoreSum M.score (sample θ)
```

Exact Lean library declaration

```
/-- Aggregate score projection from the joint floor-count rating sample. -/
def finiteChainJointFloorAverageScore
    {n : ℕ} {Rating : Type*} [Fintype Rating] [DecidableEq Rating]
    (M : FiniteRatingLDPModel (Fin n) Rating) (sampleRate : Fin n → ℝ)
    (k : ℕ) (sample : finiteChainJointFloorRatingSample Rating sampleRate k)
    (θ : Fin n) : ℝ :=
  ((floorSampleCount sampleRate θ k : ℕ) : ℝ)⁻¹ *
    finiteIidScoreSum M.score (sample θ)
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.finiteChainJointFloorRatingSample
- AppliedModelingLib.Probability.finiteIidScoreSum
- AppliedModelingLib.Probability.floorSampleCount

Recorded source-to-library screening **Verdict:** matches_approved_corrected_target

Reason: Compared cited publication:697-702, including the archived inclusive sum from 0 through n_k(theta), with the identity-bound corrected target for GJ18-AGGREGATE-INDEX-01, whose approval record expressly disclaims archival equivalence. The displayed floorSampleCount is Nat.floor(k * sampleRate theta); finiteChainJointFloorRatingSample supplies exactly Fin (floorSampleCount sampleRate theta k) ratings at each theta; finiteIidScoreSum sums M.score over that entire finite index family; and the reviewed definition multiplies that sum by the real reciprocal of precisely the same count. Thus it averages exactly n_k(theta) observations, as required by the corrected target, rather than the archival n_k(theta)+1 summands. At zero count the index family is empty, the sum is zero, and the totalized real reciprocal yields aggregate score zero, including k=0. The arbitrary real-valued score field introduces no additional premise and specializes to the source scores in [0,1], for which this average also lies in [0,1]. The displayed iidFloorSourceStateScore is a direct application of this same aggregate, and all three displayed source-claim uses pass that score unchanged into their state statistics. Those uses do not change the count, normalization, or observation indexing. This decision covers the corrected finite-chain aggregate projection only; it does not establish the source state evolution or independently certify the downstream rate claims.

Reviewer check: ☐ Matches formalized target

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

AppliedModelingLib.Probability.ratingLogMGF

Source locator: cited publication, lines 754-761

Verbatim paper-source connection

```
\begin{restatable}{theorem}{lemhatW}
    \label{lem:hatW}
    \begin{equation}
    \label{eq:rW}
    r \triangleq - \lim_{k \to \infty} \frac{1}{k} \log (1- {W}_k) = \min_{0 \leq i < M} \inf_{a \in \bbR} \left \{ g(\theta_{i+1}) I(a|\theta_{i+1}) +
    ↪   g(\theta_i) I(a|\theta_i)\right\}
    \end{equation}
    where
    $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta) = \log \sum_{y \in Y} {\rho(\theta, y | Y)} \exp( z \phi(y))$ is the log
    ↪   moment generating function of a single rating given to seller of type $\theta$.

[Next verbatim source excerpt]

\begin{lemma}
    $$     \lim_{k \to \infty}- \frac{1}{k} \log \left[\mu((x_k(\theta_1) - x_k(\theta_2)) \leq 0 | \theta_1 , \theta_2)\right] = \inf_{a \in \mathcal{R}} \left \{
    ↪   g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2) \right \}$$ where $I(a|\ell) = \sup_z \{ za - \Lambda(z|\theta) \}$, $\Lambda(z|\theta)$ is the
    ↪   log moment generating function of a single sample from $x(\theta_1)$, and $g(\theta)$ is the sampling rate. \label{lem:problessthan}
\end{lemma}

[Next verbatim source excerpt]

\begin{restatable}{lemma}{lemPkld}
    %
    \label{lem:Pk_LD}
    Given $\theta_1, \theta_2$, let $\b{P}_k(\theta_1, \theta_2) = 1 - P_k(\theta_1, \theta_2)$.  Then:
    \begin{equation}
    \label{eq:LDrate}
    - \lim_{k \to \infty} \frac{1}{k} \log \b{P}_k(\theta_1, \theta_2) = \inf_{a \in \mathcal{R}} \left \{ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2)
    ↪   \right \},
    \end{equation}
    where $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta)$ is the log moment generating function of a single rating given to
    ↪   seller of type $\theta$:
    \[ \Lambda(z|\theta) = \log \sum_{y \in Y} \rho(\theta, y | Y) \exp( z \phi(y) ). \]
```

Lean-expanded library semantic target

```
type:
{Seller : Type u_1} →
  {Rating : Type u_2} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating → Seller → ℝ → ℝ

value:
fun {Seller : Type u_1} {Rating : Type u_2} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating) (θ : Seller) (z : ℝ) =>
  M.logMGF θ z
```

Exact Lean library declaration

```
/-- Source log-MGF `Λ(z | θ) = log sum_y rho(θ,y|Y) exp(z phi(y))`. -/
def ratingLogMGF {Seller Rating : Type*} [Fintype Rating] [DecidableEq Rating]
    (M : FiniteRatingLDPModel Seller Rating) (θ : Seller) (z : ℝ) : ℝ :=
  M.logMGF θ z
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.logMGF

Recorded source-to-library screening **Verdict:** matches

Reason: Matches the displayed log-MGF definition Lambda(z|theta)=log sum_y rho(theta,y|Y) exp(z phi(y)); the expanded declaration is the finite rating-model logMGF at the selected seller and real argument.

Reviewer check: ☐ Matches source input

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

AppliedModelingLib.Probability.ratingRateFunction

Source locator: cited publication, lines 754-761

Verbatim paper-source connection

```
\begin{restatable}{theorem}{lemhatW}
	\label{lem:hatW}
	\begin{equation}
	\label{eq:rW}
	r \triangleq - \lim_{k \to \infty} \frac{1}{k} \log (1- {W}_k) = \min_{0 \leq i < M} \inf_{a \in \bbR} \left \{ g(\theta_{i+1}) I(a|\theta_{i+1}) +
	↪ g(\theta_i) I(a|\theta_i)\right\}
	\end{equation}
	where
	$I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta) = \log \sum_{y \in Y} {\rho(\theta, y | Y)} \exp( z \phi(y))$ is the log
	↪ moment generating function of a single rating given to seller of type $\theta$.

[Next verbatim source excerpt]

\begin{lemma}
	$$	\lim_{k \to \infty}- \frac{1}{k} \log \left[\mu((x_k(\theta_1) - x_k(\theta_2)) \leq 0 | \theta_1 , \theta_2)\right] = \inf_{a \in \mathcal{R}} \left \{
	↪ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2) \right \}$$ where $I(a|\ell) = \sup_z \{ za - \Lambda(z|\theta) \}$, $\Lambda(z|\theta)$ is the
	↪ log moment generating function of a single sample from $x(\theta_1)$, and $g(\theta)$ is the sampling rate. \label{lem:problessthan}
\end{lemma}

[Next verbatim source excerpt]

\begin{restatable}{lemma}{lemPkld}
	%
	\label{lem:Pk_LD}
	Given $\theta_1, \theta_2$, let $\b{P}_k(\theta_1, \theta_2) = 1 - P_k(\theta_1, \theta_2)$.  Then:
	\begin{equation}
	\label{eq:LDrate}
	- \lim_{k \to \infty} \frac{1}{k} \log \b{P}_k(\theta_1, \theta_2) = \inf_{a \in \mathcal{R}} \left \{ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2)
	↪ \right \},
	\end{equation}
	where $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta)$ is the log moment generating function of a single rating given to
	↪ seller of type $\theta$:
	\[ \Lambda(z|\theta) = \log \sum_{y \in Y} \rho(\theta, y | Y) \exp( z \phi(y) ). \]
```

Lean-expanded library semantic target

```
type:
{Seller : Type u_1} →
  {Rating : Type u_2} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] → AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating → Seller → ℝ → ℝ

value:
fun {Seller : Type u_1} {Rating : Type u_2} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating) (θ : Seller) (a : ℝ) =>
  M.rateFunction θ a
```

Exact Lean library declaration

```
/-- Source rate function `I(a | θ) = sup_z {z a - Λ(z | θ)}`. -/
def ratingRateFunction {Seller Rating : Type*} [Fintype Rating]
    [DecidableEq Rating]
    (M : FiniteRatingLDPModel Seller Rating) (θ : Seller) (a : ℝ) : ℝ :=
  M.rateFunction θ a
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.rateFunction

Recorded source-to-library screening **Verdict:** matches

Reason: Matches the archival real-valued Legendre display I(a|theta)=sup_z {za-Lambda(z|theta)}. This route remains intentionally separate from the support-safe extended-rate correction and makes no claim that the real-valued expression is valid for every threshold.

Reviewer check: ☐ Matches source input

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

AppliedModelingLib.Probability.ratingRateFunctionTop

Source locator: cited publication, lines 754-761

Verbatim paper-source connection

```
\begin{restatable}{theorem}{lemhatW}
    \label{lem:hatW}
    \begin{equation}
    \label{eq:rW}
    r \triangleq - \lim_{k \to \infty} \frac{1}{k} \log (1- {W}_k) = \min_{0 \leq i < M} \inf_{a \in \bbR} \left \{ g(\theta_{i+1}) I(a|\theta_{i+1}) +
    ↪   g(\theta_i) I(a|\theta_i)\right\}
    \end{equation}
    where
    $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta) = \log \sum_{y \in Y} {\rho(\theta, y | Y)} \exp( z \phi(y))$ is the log
    ↪   moment generating function of a single rating given to seller of type $\theta$.

[Next verbatim source excerpt]

\begin{lemma}
    $$      \lim_{k \to \infty}- \frac{1}{k} \log \left[\mu((x_k(\theta_1) - x_k(\theta_2)) \leq 0 | \theta_1 , \theta_2)\right] = \inf_{a \in \mathcal{R}} \left \{
    ↪   g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2) \right \}$$ where $I(a|\ell) = \sup_z \{ za - \Lambda(z|\theta) \}$, $\Lambda(z|\theta)$ is the
    ↪   log moment generating function of a single sample from $x(\theta_1)$, and $g(\theta)$ is the sampling rate. \label{lem:problessthan}
\end{lemma}

[Next verbatim source excerpt]

\begin{restatable}{lemma}{lemPkld}
    %
    \label{lem:Pk_LD}
    Given $\theta_1, \theta_2$, let $\b{P}_k(\theta_1, \theta_2) = 1 - P_k(\theta_1, \theta_2)$.  Then:
    \begin{equation}
    \label{eq:LDrate}
    - \lim_{k \to \infty} \frac{1}{k} \log \b{P}_k(\theta_1, \theta_2) = \inf_{a \in \mathcal{R}} \left \{ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2)
    ↪   \right \},
    \end{equation}
    where $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta)$ is the log moment generating function of a single rating given to
    ↪   seller of type $\theta$:
    \[ \Lambda(z|\theta) = \log \sum_{y \in Y} \rho(\theta, y | Y) \exp( z \phi(y) ). \]
```

Lean-expanded library semantic target

```
type:
{Seller : Type u_1} →
  {Rating : Type u_2} →
    [inst : Fintype Rating] →
      [inst_1 : DecidableEq Rating] →
        AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating → Seller → ℝ → WithTop ℝ

value:
fun {Seller : Type u_1} {Rating : Type u_2} [Fintype Rating] [DecidableEq Rating]
    (M : AppliedModelingLib.Probability.FiniteRatingLDPModel Seller Rating) (θ : Seller) (a : ℝ) =>
  M.rateFunctionTop θ a
```

Exact Lean library declaration

```
/--
Extended source rate function. Thresholds outside the finite score hull have
rate `⊤`, avoiding the real-valued `sSup` boundary that appears in unrestricted
Legendre transforms.
-/
def ratingRateFunctionTop {Seller Rating : Type*} [Fintype Rating]
    [DecidableEq Rating]
    (M : FiniteRatingLDPModel Seller Rating) (θ : Seller) (a : ℝ) :
    WithTop ℝ :=
  M.rateFunctionTop θ a
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.FiniteRatingLDPModel.rateFunctionTop

Recorded source-to-library screening **Verdict:** matches

Reason: Matches the explicitly recorded support-safe corrected rate declaration: the expanded route returns WithTop Real for the same Legendre expression, allowing infinite cost outside finite score support. This is a distinct corrected definition route, not a claim that the archival real-only display already had that codomain.

Reviewer check: ☐ Matches source input

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

Paper-specific semantic prerequisites

GJ18InformativeRatingSystems.ClarifiedSourceModel

Paper-local declaration:
GJ18InformativeRatingSystems.ClarifiedSourceModel
Source locator: cited publication, lines 19-26

Verbatim paper-source connection

```
| `GJ18-CORR-SELLER-CARRIER-2026-07-24` | `adjectivesmodel.tex:22-24,85-90,111-125` | Seller types are `theta_0 < ... < theta_{M-1}` with `M >= 2` and a uniform prior. | Makes the objective's nonzero denominator and adjacent-pair carrier explicit. |
| `GJ18-CORR-MATCH-RATES-2026-07-24` | `adjectivesmodel.tex:26-29` | For every type, `0 < g(theta) <= 1`; `n_k(theta) = floor(k g(theta))`. | Strengthens the displayed nondecreasing rate condition so every comparison has diverging sample size while preserving at most one match per period. |
| `GJ18-CORR-AGGREGATE-INDEX-2026-07-24` | `adjectivesmodel.tex:54-59` | The aggregate is the average of exactly `n_k(theta)` ratings, indexed for example by `ell = 1, ..., n_k(theta)`, with the stated zero-sample convention. | Corrects the displayed off-by-one summation endpoint. |
| `GJ18-CORR-RATING-CARRIER-2026-07-24` | `adjectivesmodel.tex:32-51` | The ordered rating scale has at least two levels; scores are strictly increasing; within-type tails decrease at displayed cutoffs; cross-type upper tails are strict only above the bottom cutoff. | Excludes the vacuous singleton scale and the impossible strict bottom-tail condition. No terminal full-support clause is added. |
| `GJ18-CORR-IID-STATE-LAW-2026-07-24` | `adjectivesmodel.tex:40,67-73` | Conditional on seller type, rating draws are iid with law `rho(theta, . | Y)` and histories are independent across sellers. At horizon `k`, the state law is the induced finite product law. | Replaces the archive's informal/ill-typed `mu_k` recurrence as the governing state-evolution clause. |
| `GJ18-CORR-ADJACENT-INDEX-2026-07-24` | `adjectivesmodel.tex:111-125; appendix_theory.tex:77-95` | The Theorem 1 minimum ranges over `0 <= i <= M-2`, exactly the valid adjacent pairs. | Corrects the endpoint that can name nonexistent `theta_M`. |
| `GJ18-CORR-EXTENDED-RATE-2026-07-24` | `adjectivesmodel.tex:111-125; appendix_theory.tex:7-8,44-53` | `I(a | theta) = sup_z {z a - Lambda(z | theta)}` is extended-real valued. The pairwise and adjacent minima use extended-real arithmetic; the final corrected-model rate is proved finite and then identified with its real representative. | Corrects the codomain required for out-of-support thresholds. |
| `GJ18-CORR-APPENDIX-DISCRETE-PROOF-2026-07-24` | `appendix_theory.tex:7-69` | The weak-inversion and `1-P_k` rate are proved directly for finite iid product laws, including the fixed-constant comparison. | Replaces the unsupported continuum-integral/Laplace transfer. |
```

[Next verbatim source excerpt]

Checked Realization

`ClarifiedSourceModel` is the complete Lean model record for these clauses. It visibly contains the seller-count, uniform-prior, match-rate, rating-scale, ordinal-tail, and iid-state-law fields. `PaperInterface.lean` exposes both the state-level extended-rate theorem and the finite-real-rate theorem for the corrected `W_k` objective.

The archived `mu_k` recurrence and unrestricted real-valued rate display are retained only as explicitly recorded archival deltas and diagnostics. No checked theorem claims to derive the corrected iid state law or corrected rate codomain from those archived expressions.

Lean-elaborated paper signature

```
field seller_type_count_pos:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate), 0 < n

field at_least_two_seller_types:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate), 2 ≤ n

field type_prior:
{n m : ℕ} →
  {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} →
    {sampleRate : Fin n → ℝ} → GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate → PMF (Fin n)

field type_prior_uniform:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate),
  self.type_prior = GJ18InformativeRatingSystems.clarifiedUniformTypePrior n self.seller_type_count_pos

field match_rate_bounds:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate),
  GJ18InformativeRatingSystems.clarifiedMatchRateBounds sampleRate

field nontrivial_rating_scale:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate),
  GJ18InformativeRatingSystems.clarifiedNontrivialRatingScale m

field ordinal_source_model:
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate),
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate

field stateLaw:
{n m : ℕ} →
  {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} →
    {sampleRate : Fin n → ℝ} →
      GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate →
        (k : ℕ) → PMF (GJ18InformativeRatingSystems.iidFloorSourceStateCarrier (Fin (m + 1)) sampleRate k)

field conditional_iid_ratings_and_independent_seller_histories:
```

```
∀ {n m : ℕ} {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (self : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate) (k : ℕ),
  self.stateLaw k = GJ18InformativeRatingSystems.iidFloorSourceStateLaw M sampleRate k
```

Direct retained dependencies

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- GJ18InformativeRatingSystems.FiniteOrdinalSourceModel
- GJ18InformativeRatingSystems.clarifiedMatchRateBounds
- GJ18InformativeRatingSystems.clarifiedNontrivialRatingScale
- GJ18InformativeRatingSystems.clarifiedUniformTypePrior
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier
- GJ18InformativeRatingSystems.iidFloorSourceStateLaw

Recorded source-to-declaration screening **Verdict:** matches

Reason: Matches the exact governing model record. The expanded structure contains the finite ordered seller carrier with at least two types, uniform prior, positive bounded matching schedule, nontrivial ordinal rating scale and tails, and the finite iid product state-law equality. Its comments and fields explicitly deny archival derivation from the printed recurrence.

Reviewer check: ☐ Matches source input

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

GJ18InformativeRatingSystems.sourceStatePairwisePk

Paper-local declaration:
GJ18InformativeRatingSystems.sourceStatePairwisePk
Source locator: cited publication, lines 721-725

Verbatim paper-source connection

```
Formally, given $\theta_1 > \theta_2$, define $P_k(\theta_1, \theta_2)$ as follows:
\begin{equation}
\label{eq:Pk}
P_k(\theta_1, \theta_2) = \mu_k(x_k(\theta_1) > x_k(\theta_2)|\theta_1, \theta_2) - \mu_k(x_k(\theta_1) < x_k(\theta_2)|\theta_1, \theta_2).
\end{equation}
```

Lean-expanded paper semantic target

```
type:
{n : ℕ} →
  {Omega : Type u_1} →
    [Fintype Omega] →
      [DecidableEq Omega] → PMF Omega → (Omega → Fin n → ℝ) → GJ18InformativeRatingSystems.finiteChainOrderedPair n → ℝ

value:
fun {n : ℕ} {Omega : Type u_1} [Fintype Omega] [DecidableEq Omega] (mu : PMF Omega) (score : Omega → Fin n → ℝ)
    (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n) =>
  (AppliedModelingLib.pmfProb mu fun (omega : Omega) =>
      score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p) >
        score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p)) -
    AppliedModelingLib.pmfProb mu fun (omega : Omega) =>
      score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p) <
        score omega (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p)
```

Direct retained dependencies

- AppliedModelingLib.pmfProb
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo

Recorded source-to-declaration screening **Verdict:** matches

Reason: Matches source equation (8): on precisely one strict high/low pair, the expanded definition subtracts the conditional probability of reversed aggregate-score order from the probability of correct strict order. It is deliberately state-level and does not turn the source recurrence into an iid law.
Reviewer check: ☐ Matches source input
If left unchecked, explain the discrepancy or uncertainty below.
Reviewer annotation

GJ18InformativeRatingSystems.sourceStateWk

Paper-local declaration:
GJ18InformativeRatingSystems.sourceStateWk
Source locator: cited publication, lines 728-733

Verbatim paper-source connection

```
We consider the problem of maximizing the following objective, a scaled version of {\em Kendall's $\tau$ rank correlation} between the estimated ranking
↪    of sellers and the true ranking:
\begin{equation}
\label{eq:objective}
W_k = \frac{2}{M(M-1)}\sum_{\theta_1 > \theta_2 \in \Theta} P_k(\theta_1, \theta_2)
\end{equation}
The coefficient ensures that ${W}_k$ remains bounded even as $M$ increases. This objective depends on the model primitives $R$ (rater behavior) and
↪    $g$ (matching rates), as well as the platform's decisions $Y$ (levels) and $\phi$ (score).
```

Lean-expanded paper semantic target

```
type:
{n : ℕ} → {Omega : Type u_1} → [Fintype Omega] → [DecidableEq Omega] → PMF Omega → (Omega → Fin n → ℝ) → ℝ

value:
fun {n : ℕ} {Omega : Type u_1} [Fintype Omega] [DecidableEq Omega] (mu : PMF Omega) (score : Omega → Fin n → ℝ) =>
  2 / (↑n * (↑n - 1)) *
    ∑ p : GJ18InformativeRatingSystems.finiteChainOrderedPair n,
      GJ18InformativeRatingSystems.sourceStatePairwisePk mu score p
```

Direct retained dependencies

- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.sourceStatePairwisePk

Recorded source-to-declaration screening **Verdict:** matches

Reason: Matches source equation (9): the expanded definition uses the displayed coefficient 2/(M(M-1)) and sums the source P_k objective over exactly the strict ordered high/low pair carrier. No additional model or convergence claim is hidden in this formula declaration.
Reviewer check: ☐ Matches source input
If left unchecked, explain the discrepancy or uncertainty below.
Reviewer annotation

Source claims

Review row 1

Source locator: cited publication, lines 1633-1635

Verbatim source input

```
\begin{lemma}
    $$      \lim_{k \to \infty}- \frac{1}{k} \log \left[\mu((x_k(\theta_1) - x_k(\theta_2)) \leq 0 | \theta_1 , \theta_2)\right] = \inf_{a \in \mathcal{R}} \left \{ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2) \right \}$$ where $I(a|\ell) = \sup_z \{ za - \Lambda(z|\theta) \}$, $\Lambda(z|\theta)$ is the log moment generating function of a single sample from $x(\theta_1)$, and $g(\theta)$ is the sampling rate. \label{lem:problessthan}
\end{lemma}

[Next verbatim source excerpt]

{\bf Sellers}.  The system consists of a unit mass of sellers, each associated with a quality \textit{type} $\theta$, which is (initially) unknown to the platform. We assume $\theta$ is drawn independently and uniformly at random from a finite and totally ordered set $\Theta$, with $|\Theta| = M$. We use $\theta_i$ to denote the $i$th element of $\Theta$ within this order, for $0\leq i < M$.

[Next verbatim source excerpt]

Formally, given $\theta_1 > \theta_2$, define $P_k(\theta_1, \theta_2)$ as follows:
\begin{equation}
\label{eq:Pk}
P_k(\theta_1, \theta_2) = \mu_k(x_k(\theta_1) > x_k(\theta_2)|\theta_1, \theta_2) - \mu_k(x_k(\theta_1) < x_k(\theta_2)|\theta_1, \theta_2).
\end{equation}
This expression captures the ``errors'' made by the ranking according to observed score.  In particular, when $\theta_1 > \theta_2$ but $x_k(\theta_1) < x_k(\theta_2)$, the aggregate score ranking swaps the ordering of sellers $\theta_1$ and $\theta_2$.  Thus, a good rating system has large $P_k(\theta_1, \theta_2)$.

We consider the problem of maximizing the following objective, a scaled version of {\em Kendall's $\tau$ rank correlation} between the estimated ranking of sellers and the true ranking:
\begin{equation}
\label{eq:objective}
W_k = \frac{2}{M(M-1)}\sum_{\theta_1 > \theta_2 \in \Theta} P_k(\theta_1, \theta_2)

[Next verbatim source excerpt]

    & = - \lim_{k \to \infty} \frac{1}{k} \log \left(1 - \frac{2}{M(M-1)}\sum_{\theta_1 > \theta_2 \in \Theta} P_k(\theta_1, \theta_2)\right)\\
    &= - \lim_{k \to \infty} \frac{1}{k} \log\frac{2}{M(M-1)} \sum_{0  \leq i < j \leq M} \b{P}_k(\theta_j, \theta_i)\\
    &= -\max_{0\leq i < j \leq M}\left( \lim_{k \to \infty} \frac{1}{k} \log \left( \b{P}_k(\theta_j, \theta_i)\right)\right)\label{eqnstep:ldpproperty}
    = \min_{0\leq i < j \leq M}\left( -\lim_{k \to \infty} \frac{1}{k} \left[\log \left(\b{P}_k(\theta_j, \theta_i)\right)\right]\right)\\
    &= \min_{0 \leq i < j \leq M} \inf_{a \in \mathcal{R}} \left \{ g(\theta_j) I(a|\theta_j) + g(\theta_i) I(a|\theta_i)\right\}\\
```

Formalized review target

The archival source above is not asserted equivalent to this formalized target.

```
For every ordered high/low seller pair in the finite ordinal IID model, with positive match rates, a nontrivial rating scale, corrected ordinal tails, and an extended-real rate convention, the source-state weak-inversion probability has an exponential-rate certificate with a finite real representative of the support-safe pairwise threshold rate.
```

Archival source anchor: cited publication, lines 1633-1635

Expanded PaperInterface specification

```
∀ {n m : ℕ} [DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)]
  (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))) (sampleRate : Fin n → ℝ),
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    (∀ (theta : Fin n), 0 < sampleRate theta) →
      1 ≤ m →
        ∀ (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n),
          ∃ (rate : ℝ),
            AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop M sampleRate
                (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p)
                (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p) =
              ↑rate ∧
              AppliedModelingLib.Probability.ExponentialRateCertificate
                (fun (k : ℕ) =>
                  GJ18InformativeRatingSystems.sourceStateScoreGapLeftTailProb
                    (GJ18InformativeRatingSystems.iidFloorSourceStateLaw M sampleRate k)
                    (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k) p)
                rate
```

Retained governing declarations

- AppliedModelingLib.Probability.ExponentialRateCertificate
- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.floorSampleCount
- AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop
- GJ18InformativeRatingSystems.FiniteOrdinalSourceModel

- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier
- GJ18InformativeRatingSystems.iidFloorSourceStateLaw
- GJ18InformativeRatingSystems.iidFloorSourceStateScore
- GJ18InformativeRatingSystems.sourceStateScoreGapLeftTailProb

Lean-elaborated claim roles

```
1. parameter: ℕ
2. parameter: ℕ
3. parameter: DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)
4. parameter: AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))
5. parameter: Fin n → ℝ
6. assumption: GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate
7. assumption: ∀ (theta : Fin n), 0 < sampleRate theta
8. assumption: 1 ≤ m
9. parameter: GJ18InformativeRatingSystems.finiteChainOrderedPair n
10. conclusion: ∃ (rate : ℝ),
 AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop M sampleRate
     (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p)
     (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p) =
    ↑rate ∧
  AppliedModelingLib.Probability.ExponentialRateCertificate
    (fun (k : ℕ) =>
     GJ18InformativeRatingSystems.sourceStateScoreGapLeftTailProb
       (GJ18InformativeRatingSystems.iidFloorSourceStateLaw M sampleRate k)
       (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k) p)
    rate
```

Lean proof endpoint (built separately)

GJ18InformativeRatingSystems.PaperInterface.source_appendix_problessthan_score_gap_rate_iid_completion

Recorded source-to-Spec screening **Verdict:** matches_approved_corrected_target

Reason: Matches the documented corrected Appendix target: the fully expanded Spec quantifies over the same finite strict high/low pair carrier and certifies the support-safe finite rate for the literal weak score-gap event under the approved finite ordinal IID model. The added model premises and extended-rate convention are exactly the recorded correction; the target does not conflate them with the archival recurrence or real-only infimum.

Reviewer check: ☐ Matches formalized target

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

Review row 2

Source locator: cited publication, lines 1670-1681

Verbatim source input

```
\begin{restatable}{lemma}{lemPkld}
    %
    \label{lem:Pk_LD}
    Given $\theta_1, \theta_2$, let $\b{P}_k(\theta_1, \theta_2) = 1 - P_k(\theta_1, \theta_2)$.  Then:
    \begin{equation}
    \label{eq:LDrate}
    - \lim_{k \to \infty} \frac{1}{k} \log \b{P}_k(\theta_1, \theta_2) = \inf_{a \in \mathcal{R}} \left \{ g(\theta_1) I(a|\theta_1) + g(\theta_2) I(a|\theta_2)
    ↪    \right \},
    \end{equation}
    where $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta)$ is the log moment generating function of a single rating given to
    ↪    seller of type $\theta$:
    \[ \Lambda(z|\theta) = \log \sum_{y \in Y} \rho(\theta, y | Y) \exp( z \phi(y) ). \]
    %
\end{restatable}

[Next verbatim source excerpt]

{\bf Sellers}.  The system consists of a unit mass of sellers, each associated with a quality \textit{type} $\theta$, which is (initially) unknown to the
↪    platform. We assume $\theta$ is drawn independently and uniformly at random from a finite and totally ordered set $\Theta$, with $|\Theta| = M$.
↪    We use $\theta_i$ to denote the $i$th element of $\Theta$ within this order, for $0\leq i < M$.

[Next verbatim source excerpt]

Formally, given $\theta_1 > \theta_2$, define $P_k(\theta_1, \theta_2)$ as follows:
\begin{equation}
\label{eq:Pk}
P_k(\theta_1, \theta_2) = \mu_k(x_k(\theta_1) > x_k(\theta_2)|\theta_1, \theta_2) - \mu_k(x_k(\theta_1) < x_k(\theta_2)|\theta_1, \theta_2).
\end{equation}
This expression captures the ``errors'' made by the ranking according to observed score.  In particular, when $\theta_1 > \theta_2$ but $x_k(\theta_1) <
↪    x_k(\theta_2)$, the aggregate score ranking swaps the ordering of sellers $\theta_1$ and $\theta_2$.  Thus, a good rating system has large
↪    $P_k(\theta_1, \theta_2)$.

We consider the problem of maximizing the following objective, a scaled version of {\em Kendall's $\tau$ rank correlation} between the estimated ranking
↪    of sellers and the true ranking:
\begin{equation}
\label{eq:objective}
W_k = \frac{2}{M(M-1)}\sum_{\theta_1 > \theta_2 \in \Theta} P_k(\theta_1, \theta_2)

[Next verbatim source excerpt]

    & = - \lim_{k \to \infty} \frac{1}{k} \log \left(1 - \frac{2}{M(M-1)}\sum_{\theta_1 > \theta_2 \in \Theta} P_k(\theta_1, \theta_2)\right)\\
    &= - \lim_{k \to \infty} \frac{1}{k} \log\frac{2}{M(M-1)} \sum_{0  \leq i < j \leq M} \b{P}_k(\theta_j, \theta_i)\\
    &= -\max_{0\leq i < j \leq M}\left( \lim_{k \to \infty} \frac{1}{k} \log \left( \b{P}_k(\theta_j, \theta_i)\right)\right)\label{eqnstep:ldpproperty}
    = \min_{0\leq i < j \leq M}\left( -\lim_{k \to \infty} \frac{1}{k} \left[\log \left(\b{P}_k(\theta_j, \theta_i)\right)\right]\right)\\
    &= \min_{0 \leq i < j \leq M} \inf_{a \in \mathcal{R}} \left \{ g(\theta_j) I(a|\theta_j) + g(\theta_i) I(a|\theta_i)\right\}\\
```

Formalized review target The archival source above is not asserted equivalent to this formalized target.

```
For every ordered high/low seller pair in the finite ordinal IID model, with positive match rates, a nontrivial rating scale, corrected ordinal tails, and an
↪    extended-real rate convention, one minus the source-state P_k objective has an exponential-rate certificate with the same finite real representative of
↪    the support-safe pairwise threshold rate.
```

Archival source anchor: cited publication, lines 1670-1681

Expanded PaperInterface specification

```
∀ {n m : ℕ} [DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)]
  (M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))) (sampleRate : Fin n → ℝ),
  GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate →
    (∀ (theta : Fin n), 0 < sampleRate theta) →
      1 ≤ m →
        ∀ (p : GJ18InformativeRatingSystems.finiteChainOrderedPair n),
          ∃ (rate : ℝ),
            AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop M sampleRate
                (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p)
                (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p) =
              ↑rate ∧
            AppliedModelingLib.Probability.ExponentialRateCertificate
              (fun (k : ℕ) =>
                1 -
                  GJ18InformativeRatingSystems.sourceStatePairwisePk
                    (GJ18InformativeRatingSystems.iidFloorSourceStateLaw M sampleRate k)
                    (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k) p)
              rate
```

Retained governing declarations

- AppliedModelingLib.Probability.ExponentialRateCertificate
- AppliedModelingLib.Probability.FiniteRatingLDPModel

- AppliedModelingLib.Probability.floorSampleCount
- AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop
- GJ18InformativeRatingSystems.FiniteOrdinalSourceModel
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.finiteChainOrderedPairHi
- GJ18InformativeRatingSystems.finiteChainOrderedPairLo
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier
- GJ18InformativeRatingSystems.iidFloorSourceStateLaw
- GJ18InformativeRatingSystems.iidFloorSourceStateScore
- GJ18InformativeRatingSystems.sourceStatePairwisePk

Lean-elaborated claim roles

```
1. parameter: ℕ
2. parameter: ℕ
3. parameter: DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)
4. parameter: AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))
5. parameter: Fin n → ℝ
6. assumption: GJ18InformativeRatingSystems.FiniteOrdinalSourceModel M sampleRate
7. assumption: ∀ (theta : Fin n), 0 < sampleRate theta
8. assumption: 1 ≤ m
9. parameter: GJ18InformativeRatingSystems.finiteChainOrderedPair n
10. conclusion: ∃ (rate : ℝ),
  AppliedModelingLib.Probability.pairwiseSellerThresholdRateTop M sampleRate
      (GJ18InformativeRatingSystems.finiteChainOrderedPairHi p)
      (GJ18InformativeRatingSystems.finiteChainOrderedPairLo p) =
    ↑rate ∧
   AppliedModelingLib.Probability.ExponentialRateCertificate
    (fun (k : ℕ) =>
      1 -
       GJ18InformativeRatingSystems.sourceStatePairwisePk
        (GJ18InformativeRatingSystems.iidFloorSourceStateLaw M sampleRate k)
        (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k) p)
    rate
```

Lean proof endpoint (built separately)

GJ18InformativeRatingSystems.PaperInterface.source_appendix_pk_ld_complement_rate_iid_completion

Recorded source-to-Spec screening **Verdict:** matches_approved_corrected_target

Reason: Matches the documented corrected Appendix target: the fully expanded Spec quantifies only over the finite strict high/low pair carrier, retains the finite ordinal and positive-rate/nontrivial-scale premises, and certifies the finite representative of the support-safe pairwise rate for one minus the source-state P_k objective. The source's omitted pair qualifier and real-only rate display are explicit approved corrections, not hidden derivations.

Reviewer check: ☐ Matches formalized target

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation

Review row 3

Source locator: cited publication, lines 754-764

Verbatim source input

```
\begin{restatable}{theorem}{lemhatW}
    \label{lem:hatW}
    \begin{equation}
    \label{eq:rW}
    r \triangleq - \lim_{k \to \infty} \frac{1}{k} \log (1- {W}_k) = \min_{0 \leq i < M} \inf_{a \in \bbR} \left \{ g(\theta_{i+1}) I(a|\theta_{i+1}) +
    ↪   g(\theta_i) I(a|\theta_i)\right\}
    \end{equation}
    where
    $I(a|\theta) = \sup_z \{ za - \Lambda(z|\theta) \}$, and $\Lambda(z|\theta) = \log \sum_{y \in Y} {\rho(\theta, y | Y)} \exp( z \phi(y))$ is the log
    ↪   moment generating function of a single rating given to seller of type $\theta$.
%
%
\end{restatable}
```

Formalized review target

The archival source above is not asserted equivalent to this formalized target.

```
For the corrected finite ordinal IID model with at least two uniformly distributed seller types, positive at-most-one-per-period match rates, a nontrivial
↪   ordinal rating scale, and valid adjacent pairs 0 <= i <= M-2, one minus the state-level W_k objective has a finite real exponential rate equal to the
↪   finite representative of the support-safe adjacent-pair rate.
```

Archival source anchor: cited publication, lines 754-764

Expanded PaperInterface specification

```
∀ {n m : ℕ} [DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)]
  {M : AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))} {sampleRate : Fin n → ℝ}
  (model : GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate),
  ∃ (rate : ℝ),
    model.correctedTheorem1Rate = ↑rate ∧
      AppliedModelingLib.Probability.HasExponentialRate
        (fun (k : ℕ) =>
          1 -
            GJ18InformativeRatingSystems.sourceStateWk (model.stateLaw k)
              (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k))
        rate
```

Retained governing declarations

- AppliedModelingLib.Probability.FiniteRatingLDPModel
- AppliedModelingLib.Probability.HasExponentialRate
- AppliedModelingLib.Probability.floorSampleCount
- GJ18InformativeRatingSystems.ClarifiedSourceModel
- GJ18InformativeRatingSystems.ClarifiedSourceModel.correctedTheorem1Rate
- GJ18InformativeRatingSystems.finiteChainOrderedPair
- GJ18InformativeRatingSystems.iidFloorSourceStateCarrier
- GJ18InformativeRatingSystems.iidFloorSourceStateScore
- GJ18InformativeRatingSystems.sourceStateWk

Lean-elaborated claim roles

```
1. parameter: ℕ
2. parameter: ℕ
3. parameter: DecidableEq (GJ18InformativeRatingSystems.finiteChainOrderedPair n)
4. parameter: AppliedModelingLib.Probability.FiniteRatingLDPModel (Fin n) (Fin (m + 1))
5. parameter: Fin n → ℝ
6. parameter: GJ18InformativeRatingSystems.ClarifiedSourceModel M sampleRate
7. conclusion: ∃ (rate : ℝ),
  model.correctedTheorem1Rate = ↑rate ∧
    AppliedModelingLib.Probability.HasExponentialRate
      (fun (k : ℕ) =>
        1 -
          GJ18InformativeRatingSystems.sourceStateWk (model.stateLaw k)
            (GJ18InformativeRatingSystems.iidFloorSourceStateScore M sampleRate k))
      rate
```

Lean proof endpoint (built separately)

```
GJ18InformativeRatingSystems.PaperInterface.author_approved_corrected_model_theorem1_finite_real_rate
```

Recorded source-to-Spec screening **Verdict:** matches_approved_corrected_target

Reason: Matches the documented corrected target: the fully expanded Spec takes the complete ClarifiedSourceModel and concludes that its support-safe adjacent-pair rate has a finite real representative governing the exponential rate of one minus the state-level W_k objective. This is exactly the approved finite-chain/IID replacement and does not claim that the archive's recurrence or unrestricted Real display derives it.

Reviewer check: ☐ Matches formalized target

If left unchecked, explain the discrepancy or uncertainty below.

Reviewer annotation