Gaussian Processes and Kernel Ridge Regression

Theoretical Connections, Hyperparameter Tuning, and Experiments

Team 10: Leonardo Cartesegna, Chandrasekhara Devarakonda, Giulia Scagliarini

MATH-412: Statistical Machine Learning

December 9, 2025

Outline

Table of Contents

Context and Motivation

- Focus: the link between **Gaussian process (GP) regression** and **kernel ridge regression (KRR)**.
- Both methods are built from a positive definite kernel $k(\cdot, \cdot)$ and yield very similar predictors.
- However:
  - GP regression is a **Bayesian probabilistic** model (predictive mean + uncertainty).
  - KRR is a **regularization/optimization** method in an RKHS (point estimate).

Project Objectives

- **Theoretical comparison**
  - Show the equivalence between GP posterior mean and KRR predictor under a parameter mapping.
  - Explain the RKHS viewpoint and the MAP interpretation.
- **Hyperparameter selection**
  - GP: **marginal likelihood** (type-II ML / evidence maximization).
  - KRR: **cross-validation** (K-fold).
  - Also test **CV for GPs** with two criteria: MSE vs NLPD.
- **Empirical study**
  - Static financial dataset (portfolio performance), $n = 63$, $D = 6$.
  - Same kernel, same preprocessing, same split across all methods.
  - Stay in an **interpolation** regime (avoid unjustified time extrapolation).

Table of Contents

Supervised Regression Setup

- Training set

$$\mathcal{D} = \{(x_i, y_i) \mid i = 1, \ldots, n\}, \quad x_i \in \mathbb{R}^D, \; y_i \in \mathbb{R}.$$

- Inputs stacked into a design matrix (Python/NumPy convention):

$$X \in \mathbb{R}^{n \times D}, \quad \text{row } i \text{ is } x_i^\top.$$

- Targets:

$$\mathrm{y} = (y_1, \ldots, y_n)^\top \in \mathbb{R}^n.$$

- Goal: for a new input x_*, infer y_* (and the latent value $f(x_*)$).

Observation Generation Model

- Latent function with additive Gaussian noise:

$$y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma_n^2) \text{ i.i.d.}$$

- Conditional observation model:

$$y_i \mid f(x_i) \sim \mathcal{N}(f(x_i), \sigma_n^2).$$

- Two perspectives:
  - **GP regression:** posterior distribution over functions f (Bayesian approach).
  - **KRR:** single function $\hat{f}$ minimizing regularized risk in an RKHS $\mathcal{H}_k$.

Table of Contents

Gaussian Process Prior

- GP definition: any finite subset is jointly Gaussian:

$$f \sim \mathcal{GP}(m(\cdot), k(\cdot, \cdot)).$$

- We assume $m(x) = 0$ for simplicity.
- Kernel $k_\theta(x, x')$ with hyperparameters θ (e.g. ℓ, σ_f^2).
- Prior over latent values at training inputs:

$$\mathrm{f} = (f(x_1), \ldots, f(x_n))^\top \sim \mathcal{N}(0, K), \quad K_{ij} = k(x_i, x_j).$$

Likelihood and Joint Distribution

- Likelihood (Gaussian noise):

$$p(\mathrm{y} \mid \mathrm{f}) = \mathcal{N}(\mathrm{f}, \sigma_n^2 I).$$

- Marginalizing f:

$$\mathrm{y} \sim \mathcal{N}(0, K_y), \qquad K_y = K + \sigma_n^2 I.$$

- For a test input x_*, let $f_* = f(x_*)$. Joint prior:

$$\begin{bmatrix} \mathrm{f} \\ f_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} K & \mathrm{k}_* \\ \mathrm{k}_*^\top & k_{**} \end{bmatrix} \right),$$

where $\mathrm{k}_* = k(X, x_*) \in \mathbb{R}^n$, $k_{**} = k(x_*, x_*)$.

GP Posterior and Predictions

$$f_* \mid x_*, X, \mathrm{y} \sim \mathcal{N}\big(m(x_*), \mathrm{Var}[f_* \mid \mathcal{D}]\big),$$

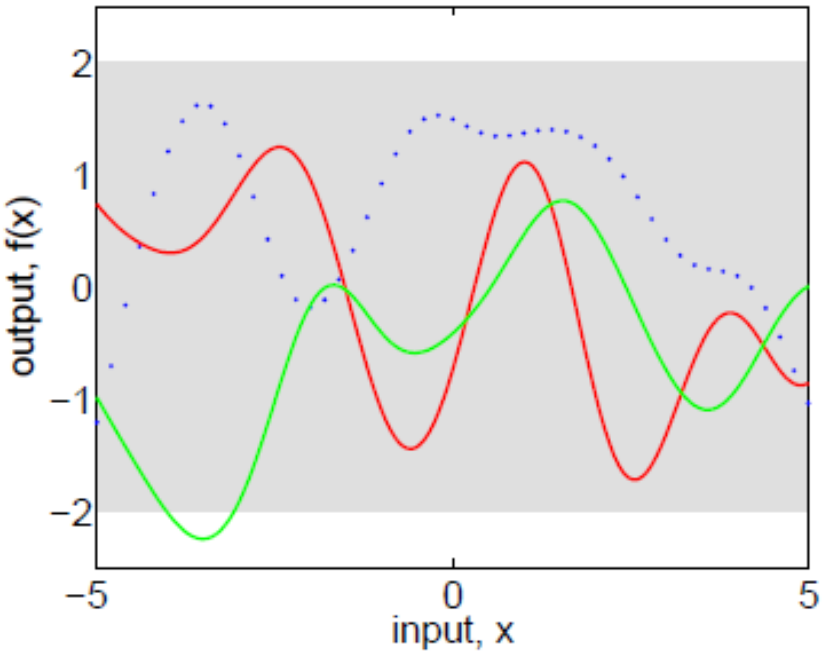
with

$$m(x_*) = \mathrm{k}_*^\top K_y^{-1} \mathrm{y},$$
$$\mathrm{Var}[f_* \mid \mathcal{D}] = k_{**} - \mathrm{k}_*^\top K_y^{-1} \mathrm{k}_*.$$

- Posterior mean $m(x_*)$: point prediction for the latent function under squared loss.
- Posterior variance: uncertainty for $f(x_*)$.
- Predictive distribution for the *noisy* target:

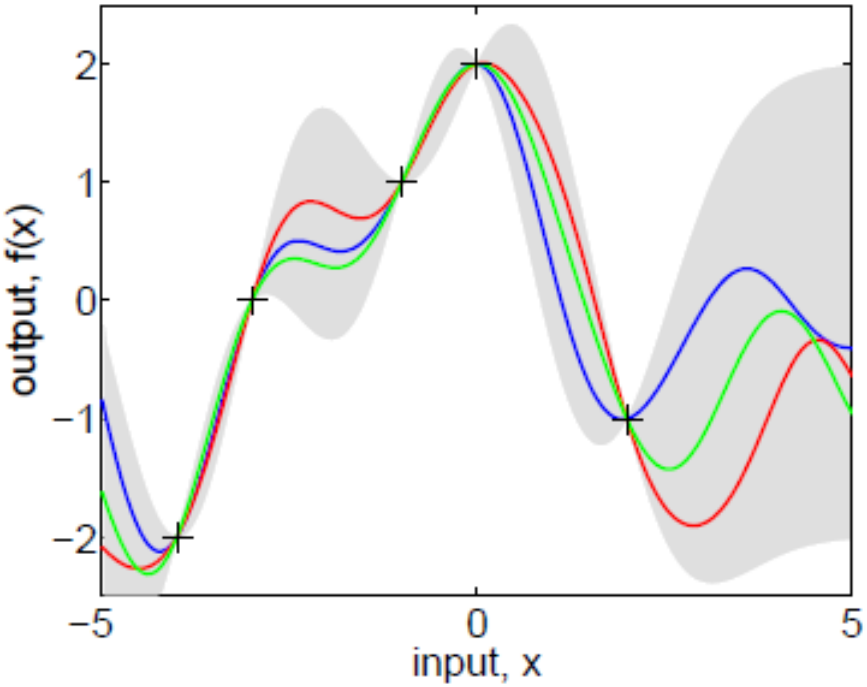
$$y_* \mid x_*, \mathcal{D} \sim \mathcal{N}\big(m(x_*),\ \mathrm{Var}[f_* \mid \mathcal{D}] + \sigma_n^2\big).$$

GP Prior and Posterior



Prior

Before seeing data, the prior encodes smoothness and amplitude via k.



Posterior

Conditioning on $\mathcal{D}$ shrinks uncertainty near training inputs and shapes the mean.

Inference with the GP Posterior

- GP yields a full posterior distribution, not only a single $\hat{f}$.
- Decision-theoretic point predictions depend on a loss $\mathcal{L}$:
  - Squared loss $\Rightarrow$ posterior mean.
  - Absolute loss $\Rightarrow$ posterior median.
- For Gaussian posteriors, mean = median.
- The key extra object is calibration: uncertainty bands and predictive densities.

Table of Contents

Kernel Ridge Regression Objective

- Let k be a PD kernel with RKHS $\mathcal{H}_k$.
- KRR solves

$$\min_{f \in \mathcal{H}_k} \left\{ \frac{1}{2n} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \frac{\lambda}{2} \|f\|_{\mathcal{H}_k}^2 \right\}.$$

- $\|f\|_{\mathcal{H}_k}$ acts as a smoothness/complexity measure; λ controls the bias–variance trade-off.

Representer Theorem and Finite-Dimensional Form

- Representer theorem:

$$f(x) = \sum_{i=1}^{n} \alpha_i \, k(x_i, x).$$

- Let $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^\top$, then $\mathsf{f} = K\boldsymbol{\alpha}$.
- Finite-dimensional problem:

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^n} \frac{1}{2n} \|K\boldsymbol{\alpha} - \mathsf{y}\|_2^2 + \frac{\lambda}{2} \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}.$$

Solution of Kernel Ridge Regression

- Closed form:

$$\boldsymbol{\alpha}^* = (K + \lambda n I)^{-1} \mathrm{y}.$$

- Predictor:

$$f_{\mathrm{KRR}}(x_*) = \mathrm{k}_*^\top (K + \lambda n I)^{-1} \mathrm{y}.$$

- Compare to GP posterior mean:

$$m(x_*) = \mathrm{k}_*^\top (K + \sigma_n^2 I)^{-1} \mathrm{y}.$$

Table of Contents

What the representer theorem does (and does not) explain

- The representer theorem gives the finite-dimensional form

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i \, k(x_i, \cdot).$$

- It does *not* by itself explain why the RKHS penalty matches GP regression.
- Key principle: **KRR is the MAP estimator under a GP prior with Gaussian noise.**

MAP for latent training values

Let $\mathrm{f} = (f(x_1), \ldots, f(x_n))^\top$ and $K_{ij} = k(x_i, x_j)$. Under the GP model:

$$p(\mathrm{y} \mid \mathrm{f}) = \mathcal{N}(\mathrm{y} \mid \mathrm{f}, \sigma_n^2 I), \qquad p(\mathrm{f}) = \mathcal{N}(\mathrm{f} \mid 0, K).$$

Dropping constants, the negative log-posterior is

$$-\log p(\mathrm{f} \mid X, \mathrm{y}) = \frac{1}{2\sigma_n^2} \|\mathrm{y} - \mathrm{f}\|_2^2 + \frac{1}{2} \mathrm{f}^\top K^{-1} \mathrm{f}.$$

Thus the MAP estimator solves

$$\mathrm{f}_{\mathrm{MAP}} = \arg\min_{\mathrm{f}} \left\{ \frac{1}{2\sigma_n^2} \|\mathrm{y} - \mathrm{f}\|_2^2 + \frac{1}{2} \mathrm{f}^\top K^{-1} \mathrm{f} \right\}.$$

Posterior Gaussian $\Rightarrow$ MAP = posterior mean (same linear system)

- Gaussian likelihood + Gaussian prior $\Rightarrow$ Gaussian posterior.
- For a Gaussian, **mode = mean**, hence

$$\mathrm{f}_{\mathrm{MAP}} = \mathbb{E}[\mathrm{f} \mid X, \mathrm{y}].$$

- First-order optimality condition:

$$\frac{1}{\sigma_n^2}(\mathrm{f} - \mathrm{y}) + K^{-1}\mathrm{f} = 0 \quad \Longrightarrow \quad (K + \sigma_n^2 I)\mathrm{f} = K\mathrm{y},$$

so

$$\mathrm{f}_{\mathrm{MAP}} = K(K + \sigma_n^2 I)^{-1}\mathrm{y}.$$

From the GP quadratic penalty to the RKHS norm penalty

- For $f(\cdot) = \sum_{i=1}^{n} \alpha_i k(x_i, \cdot)$, we have $\mathsf{f} = K\boldsymbol{\alpha}$.
- RKHS norm identity:

$$\|f\|_{\mathcal{H}_k}^2 = \boldsymbol{\alpha}^\top K \boldsymbol{\alpha}.$$

- If K is invertible, $\boldsymbol{\alpha} = K^{-1}\mathsf{f}$ and

$$\boldsymbol{\alpha}^\top K \boldsymbol{\alpha} = \mathsf{f}^\top K^{-1} \mathsf{f}.$$

KRR emerges by rescaling: parameter identification

- Substitute $\mathrm{f} = K\boldsymbol{\alpha}$ into the MAP objective:

$$\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2\sigma_n^2} \|\mathrm{y} - K\boldsymbol{\alpha}\|_2^2 + \frac{1}{2}\boldsymbol{\alpha}^\top K\boldsymbol{\alpha} \right\}.$$

- Multiply by σ_n^2/n (no change in minimizer):

$$\min_{\boldsymbol{\alpha}} \left\{ \frac{1}{2n} \|\mathrm{y} - K\boldsymbol{\alpha}\|_2^2 + \frac{\lambda}{2}\boldsymbol{\alpha}^\top K\boldsymbol{\alpha} \right\}, \qquad \lambda = \frac{\sigma_n^2}{n}.$$

- Hence the linear systems match under $\sigma_n^2 = \lambda n$:

$$(K + \lambda n I)\boldsymbol{\alpha} = \mathrm{y} \quad \Longleftrightarrow \quad (K + \sigma_n^2 I)\boldsymbol{\alpha} = \mathrm{y}.$$

Conclusion: same predictor, different interpretation

- Under $\sigma_n^2 = \lambda n$, GP posterior mean equals KRR predictor:

$$f_{\mathrm{KRR}}(x_*) = \mathrm{k}_*^\top (K + \lambda n I)^{-1} \mathrm{y} = \mathrm{k}_*^\top (K + \sigma_n^2 I)^{-1} \mathrm{y} = m(x_*).$$

- Interpretation:
  - **KRR:** point estimator (regularized risk minimization), equivalently a **MAP** estimate.
  - **GP:** full posterior over functions; provides uncertainty and predictive densities.

Table of Contents

Parameters/Hyperparameters

- Hyperparameters control the prior and the noise model:
  - Kernel parameters: length-scale ℓ, signal variance σ_f^2.
  - Noise variance σ_n^2 (GP).
  - Regularization λ (KRR), with mapping $\lambda = \sigma_n^2/n$.
- Question: **How do we select them?**
  - GP: maximize marginal likelihood $p(\mathrm{y} \mid X, \theta)$.
  - CV: choose hyperparameters by estimated out-of-sample performance.
  - For probabilistic models, a natural CV objective is **NLPD** (uses mean + variance), not only MSE.

Example: Squared Exponential (RBF) Kernel

- RBF / squared exponential:

$$k(x_i, x_j) = \sigma_f^2 \exp\left(-\frac{1}{2\ell^2}\|x_i - x_j\|^2\right).$$

- Hyperparameters:
  - σ_f^2: amplitude (vertical scale).
  - ℓ: smoothness length-scale (how quickly correlations decay).
  - σ_n^2: observation noise (GP) / diagonal shift (KRR via mapping).

GP: Marginal Likelihood

- Under the GP model:

$$\mathrm{y} \sim \mathcal{N}(0, K_y), \quad K_y = K + \sigma_n^2 I.$$

- Log marginal likelihood:

$$\log p(\mathrm{y} \mid X, \theta) = -\frac{1}{2}\mathrm{y}^\top K_y^{-1}\mathrm{y} - \frac{1}{2}\log |K_y| - \frac{n}{2}\log 2\pi.$$

- Interpretation:
  - Data fit term $-\frac{1}{2}\mathrm{y}^\top K_y^{-1}\mathrm{y}$.
  - Complexity penalty $-\frac{1}{2}\log |K_y|$ (Occam's razor).

Cross-Validation Objectives: MSE vs NLPD

- Point prediction objective (standard):

$$\text{MSE} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} (y_i - \hat{y}_i)^2.$$

- Probabilistic objective (for GP regression):

$$\text{NLPD} = -\frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \log \mathcal{N}\big(y_i \mid \mu_i,\ \sigma_i^2\big),$$

  where in a GP we use $\sigma_i^2 = \text{Var}[f(x_i) \mid \mathcal{D}_{\text{train}}] + \sigma_n^2$.
- MSE ignores uncertainty calibration; NLPD penalizes both bad means and miscalibrated variances.

How We Tune Hyperparameters in This Project

- **GP-ML**: minimize negative log marginal likelihood (L-BFGS-B) over ℓ, σ_f, σ_n.
- **GP-CV**: grid-search 5-fold CV over ℓ, σ_f, σ_n using two selection rules:
  - select by **CV-NLPD** (probabilistic),
  - select by **CV-MSE** (point prediction).
- **KRR-CV**: grid-search 5-fold CV over ℓ, σ_f and λ (implemented via diagonal shift $\lambda_{\text{eff}} = \lambda n$).

Table of Contents

Interpolation vs Extrapolation for GPs

- For stationary kernels and zero mean, far from training inputs:
  - $k_* \to 0$,
  - $m(x_*) \to 0$ (prior mean),
  - $\mathrm{Var}[f_* | \mathcal{D}] \to k_{**}$ (prior variance).
- Without strong prior structure, long-range extrapolation is unreliable (especially for finance).

Implications for This Project

- We avoid time extrapolation (e.g. future prices).
- Dataset is static and cross-sectional:
  - Inputs: portfolio concept weights (6D).
  - Output: normalized annual return.
- We evaluate interpolation in feature space (portfolio rules).

Table of Contents

Dataset: Portfolio Performance

- Each observation corresponds to a portfolio construction rule.
- Total $n = 63$ portfolios, input dimension $D = 6$.
- Inputs $x_i \in \mathbb{R}^6$: weights on six stock-selection concepts.
- Output y_i: normalized annual return.

Preprocessing and Splits

- Standardize inputs X using training set statistics.
- Standardize target y using training mean and std (for tuning); report final metrics on original scale.
- Train/test split:
  - 44 training portfolios, 19 test portfolios (70%/30%).
- Metrics:
  - Test MSE (original scale).
  - Mean NLPD (GP models; predictive variance includes σ_n^2).

Models Compared

- Common kernel: RBF

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{\|x - x'\|^2}{2\ell^2}\right).$$

- **GP-ML**: ℓ, σ_f, σ_n by marginal likelihood.
- **GP-CV (NLPD)**: grid-search 5-fold CV; select by mean CV-NLPD.
- **GP-CV (MSE)**: same grid-search; select by mean CV-MSE.
- **KRR-CV**: 5-fold CV (MSE) over ℓ, σ_f, λ via $\lambda_{\text{eff}} = \lambda n_{\text{train}}$.

Table of Contents

Learned Hyperparameters (Standardized Space)

Model	ℓ	σ_f	σ_n	λ (if KRR)
GP-ML	2.8143	1.2636	0.08823	–
GP-CV (NLPD)	2.6724	1.2034	0.09203	–
GP-CV (MSE)	3.2931	1.2241	0.03692	–
KRR-CV	3.2931	1.2241	–	3.098×10^{-5}

- GP-CV(MSE) and KRR-CV select identical (ℓ, σ_f) and match the GP–KRR mapping.
- With $n_{\text{train}} = 44$, GP-CV(MSE) has

$$\frac{\sigma_n^2}{n_{\text{train}}} = \frac{(0.03692)^2}{44} \approx 3.10 \times 10^{-5} \approx \lambda_{\text{KRR}},$$

and $\lambda_{\text{eff}} = \sigma_n^2 \approx 1.363 \times 10^{-3}$.

Quantitative Results (Test Set and CV)

Model	Test MSE	Mean NLPD	CV MSE (scaled y)	CV NLPD (scaled y)
GP-ML	1.822×10^{-3}	-1.780	–	–
GP-CV (NLPD)	1.907×10^{-3}	-1.820	0.1927	0.1663
GP-CV (MSE)	1.813×10^{-3}	-0.108	0.1702	0.9790
KRR-CV	1.813×10^{-3}	–	0.1702	–

- Point accuracy (MSE): all methods are very close ($\approx 1.8 \times 10^{-3}$).
- Calibration (NLPD): selecting GP hyperparameters by MSE alone can strongly degrade predictive density quality.

Calibration: GP-ML vs GP-CV (NLPD) vs GP-CV (MSE)

- **GP-ML**:
  - $\sigma_n \approx 0.088$, mean NLPD ≈ -1.78.
  - Marginal likelihood trades off fit and complexity and tends to yield calibrated uncertainty.
- **GP-CV (NLPD)**:
  - $\sigma_n \approx 0.092$, mean NLPD ≈ -1.82 (best here).
  - Directly optimizes predictive density, so it rewards calibrated variances.
- **GP-CV (MSE)**:
  - $\sigma_n \approx 0.037$, mean NLPD ≈ -0.11.
  - Good MSE, but overly small σ_n can make the model **overconfident** and hurt NLPD.

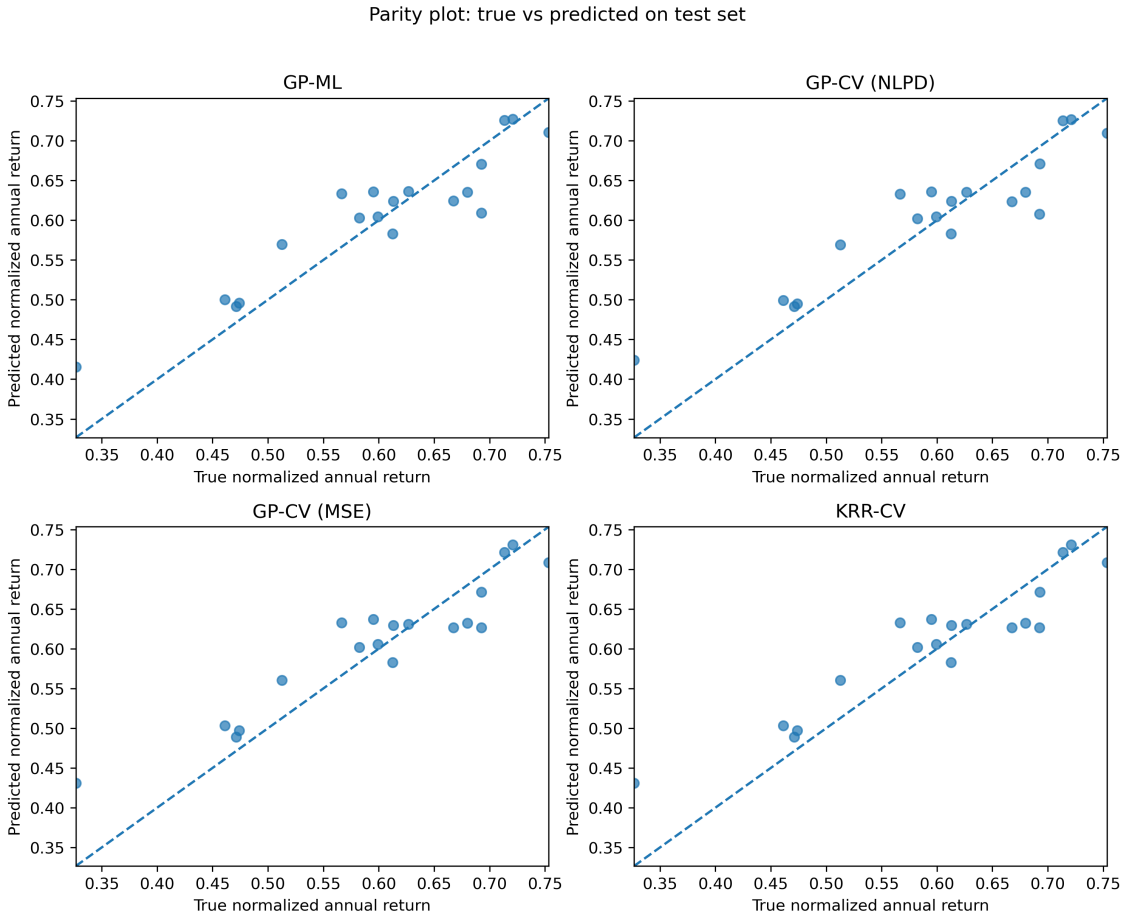
GP vs KRR Predictive Means

- Comparing GP-ML posterior mean vs KRR-CV predictor on standardized y:

$$\text{mean}\,|\Delta| \approx 3.04 \times 10^{-2},$$
$$\max|\Delta| \approx 1.24 \times 10^{-1}.$$

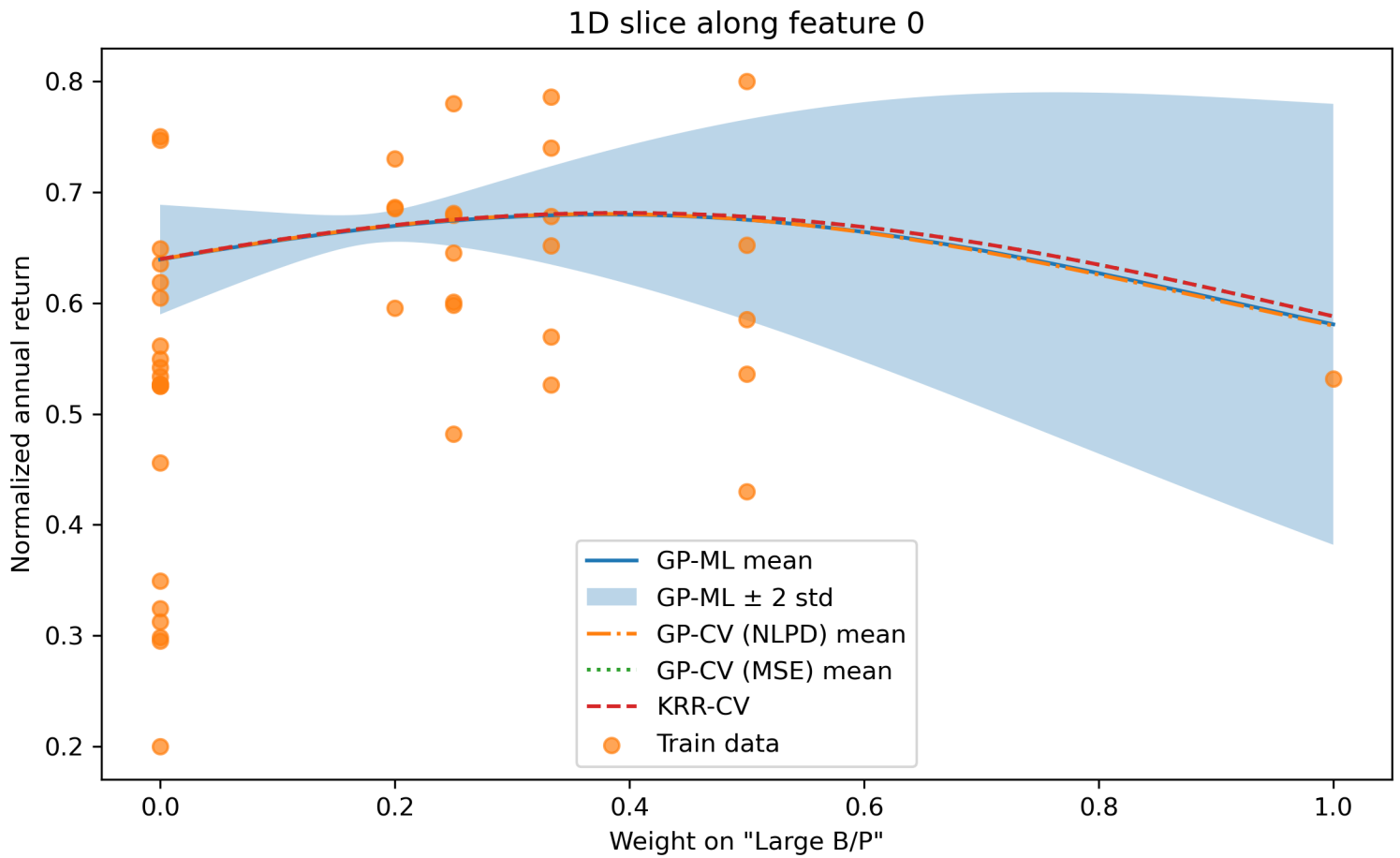
- Differences remain small given that GP-ML hyperparameters are chosen by evidence maximization, while KRR is tuned by MSE-CV.
- When GP is tuned by MSE-CV, its σ_n^2 matches KRR's diagonal shift exactly (via $\lambda_{\text{eff}} = \sigma_n^2$).

Parity Plot on Test Set



- True vs predicted normalized annual return for GP-ML, GP-CV (NLPD), GP-CV (MSE), and KRR-CV.
- All methods lie close to the diagonal, consistent with similar test MSE.

1D Slice Along a Portfolio Weight



- Vary one feature; fix others at their mean (in standardized space).
- GP-ML mean and KRR-CV curve are close; GP also provides uncertainty bands.
- Uncertainty widens away from dense regions of training data.

Table of Contents

Discussion

- **Predictive equivalence (theory):**
  - GP posterior mean = KRR predictor under $\sigma_n^2 = \lambda n$.
  - Empirically, GP-CV(MSE) and KRR-CV match the mapping almost exactly.
- **Point accuracy vs calibration:**
  - MSE is similar across methods, so point prediction alone does not distinguish them here.
  - NLPD separates GP choices: selecting by MSE can produce overconfident predictive variances.
  - Selecting by NLPD (or using marginal likelihood) better respects the probabilistic objective.
- **Interpretation difference:**
  - GP: posterior over functions + uncertainty; principled evidence-based tuning.
  - KRR: deterministic predictor; no native predictive variance (without extra machinery).

Conclusion

- We compared **GP regression** and **KRR**:
  - same kernel, same data, shared algebraic structure,
  - different interpretations and tuning principles.
- Empirical takeaways:
  - Test MSE is nearly identical across GP-ML, GP-CV, and KRR-CV on this dataset.
  - Calibration differs substantially: GP-ML and GP-CV(NLPD) yield strong NLPD; GP-CV(MSE) does not.
  - KRR matches the GP-CV(MSE) solution via $\lambda \approx \sigma_n^2/n$, confirming the theory.
- Bottom line: **GP = Bayesian KRR** (posterior mean as MAP/regularized solution) plus uncertainty quantification and evidence-based model selection.

References

C. E. Rasmussen and C. K. I. Williams,
Gaussian Processes for Machine Learning,
MIT Press, 2006.

Course notes,
MATH-412: Kernel Methods (Lecture 7b).