Covariate Shift by Kernel Mean Matching

Presented by Javier Zapata

- **Definition 16 (kernel)**

  - A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a positive semidefinite kernel (or more simply, a kernel) iff for every finite set of points $x_1, \ldots, x_n \in \mathcal{X}$, the **kernel matrix** $K \in \mathbb{R}^{n \times n}$ defined by $K_{ij} = k(x_i, x_j)$ is positive semidefinite.

- Linear Kernel: $k(x, x') = \langle x, x' \rangle$
- Polynomial Kernel: $k(x, x') = (1 + \langle x, x' \rangle)^p$
- Gaussian Kernel: $k(x, x') = \exp\left(\frac{-\|x - x'\|_2^2}{2\sigma^2}\right)$

- General principles for checking kernels:
  - $f: \mathcal{X} \to \mathbb{R}$, $k(x, x') = f(x) f(x')$
  - Sum: $k(x, x') = k_1(x, x') + k_2(x, x')$
  - Product: $k(x, x') = k_1(x, x')\ k_2(x, x')$
  - Mapping between spaces: Let A: $\mathcal{X} \to \widetilde{\mathcal{X}}$, and $\tilde{k}(\cdot,\cdot)$ a kernel on $\widetilde{\mathcal{X}}$. Then, $\mathrm{k}(\mathrm{x}, \mathrm{x}') = \tilde{k}(A(x), A(x'))$ is a kernel on $\mathcal{X}$

- **Kernel trick**:= For any algorithm that depends on $x^T x'$, replace it with $k(x, x')$, since kernels define a measure of similarity between x and x'

- **Definition 18 (feature map)**
  - Given a Hilbert space $\mathcal{H}$, a **feature map** $\phi : \mathcal{X} \to \mathcal{H}$ takes inputs $x \in \mathcal{X}$ to infinite feature vectors $\phi(x) \in \mathcal{H}$.

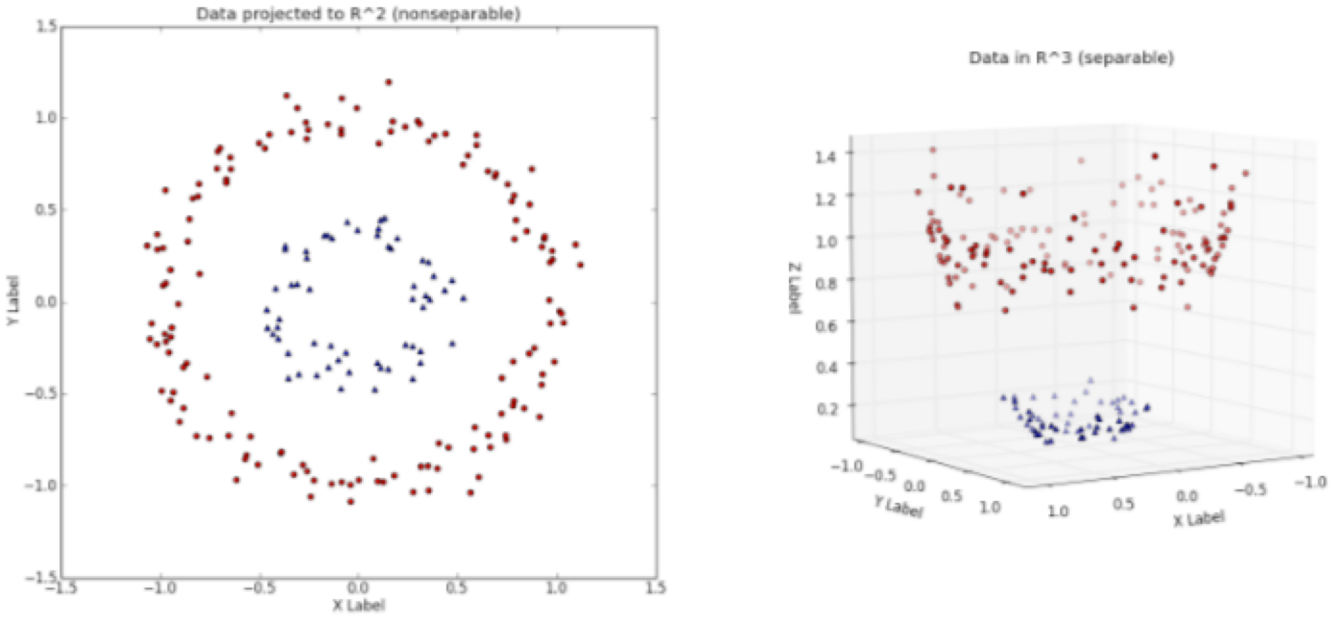


Figure 5: (Left) A dataset in $\mathbb{R}^2$, not linearly separable. (Right) The same dataset transformed by the transformation: $[x_1, x_2] = [x_1, x_2, {x_1}^2 + {x_2}^2]$.

- **Definition 17 (Hilbert space)**
  - A Hilbert space $\mathcal{H}$ is an complete[11] vector space with an inner product $\langle \cdot, \cdot \rangle : \mathcal{H} \times \mathcal{H} \to \mathbb{R}$ that satisfies the following properties:
    * Symmetry: $\langle f, g \rangle = \langle g, f \rangle$
    * Linearity: $\langle \alpha_1 f_1 + \alpha_2 f_2, g \rangle = \alpha_1 \langle f_1, g \rangle + \alpha_2 \langle f_2, g \rangle$
    * Positive definiteness: $\langle f, f \rangle \geq 0$ with equality only if $f = 0$

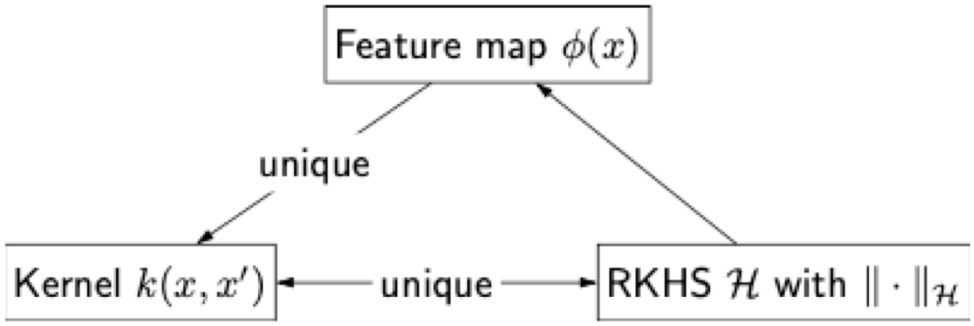
    The inner product gives us a norm: $\|f\|_{\mathcal{H}} \stackrel{\text{def}}{=} \sqrt{\langle f, f \rangle}$.

  - For example, consider $\mathcal{H} = L^2([0, 1])$. Recall that every $f \in \mathcal{H}$ is actually an equivalence class over functions which differ on a measure zero set, which means pointwise evaluations $f(x)$ at individual x's is not even defined.
  - This is highly distressing given that the whole point is to learn an f for the purpose of doing pointwise evaluations (a.k.a. prediction)!
  - RKHSes remedy this problem by making pointwise evaluations really nice, as we'll see.

Reproducing Kernel Hilbert Spaces (RKHS)

$\mathcal{H}$ a Hilbert space of $\mathbb{R}$-valued functions on non-empty set $\mathcal{X}$. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a reproducing kernel of $\mathcal{H}$, and $\mathcal{H}$ is a reproducing kernel Hilbert space, if

- $\forall x \in \mathcal{X}, \quad k(\cdot, x) \in \mathcal{H}$,
- $\forall x \in \mathcal{X}, \forall f \in \mathcal{H}, \quad \langle f(\cdot), k(\cdot, x) \rangle_{\mathcal{H}} = f(x)$ (the reproducing property).



Another RKHS definition:

Define δ_x to be the operator of evaluation at x, i.e.

$$\delta_x f = f(x) \quad \forall f \in \mathcal{H},\ x \in \mathcal{X}.$$

Definition (Reproducing kernel Hilbert space)

$\mathcal{H}$ is an RKHS if the evaluation operator δ_x is bounded: $\forall x \in \mathcal{X}$ there exists $\lambda_x \geq 0$ such that for all $f \in \mathcal{H}$,

$$|f(x)| = |\delta_x f| \leq \lambda_x \|f\|_{\mathcal{H}}$$

$\Longrightarrow$ two functions identical in RHKS norm agree at every point:

$$|f(x) - g(x)| = |\delta_x (f - g)| \leq \lambda_x \|f - g\|_{\mathcal{H}} \quad \forall f, g \in \mathcal{H}.$$

Universality (general purpose kernel)

- **Definition 23 (universal kernel)**
  - Let $\mathcal{X}$ be a locally compact Hausdorff space (e.g., $\mathbb{R}^b$ or any discrete set, but not infinite-dimensional spaces in general).
  - Let $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a kernel.
  - We say that k is a **universal kernel** (specifically, a c_0-universal kernel) iff the RKHS $\mathcal{H}$ with reproducing kernel k is dense in $C_0(\mathcal{X})$, the set of all continuous bounded functions on $\mathcal{X}$ (with respect to the uniform norm). In other words, for any function $f \in C_0(\mathcal{X})$ and $\epsilon > 0$, there exists some $g \in \mathcal{H}$ such that $\sup_{x \in \mathcal{X}} \|f - g\| \leq \epsilon$.
- Example:
  - Gaussian kernel is universal

Dataset Shift

- Training Set: $\left\{(x_1^{\mathrm{tr}}, y_1^{\mathrm{tr}}), \ldots, (x_{n_{\mathrm{tr}}}^{\mathrm{tr}}, y_{n_{\mathrm{tr}}}^{\mathrm{tr}})\right\} \subseteq \mathcal{X} \times \mathcal{Y}$
- Test Set: $\left\{(x_1^{\mathrm{te}}, y_1^{\mathrm{te}}), \ldots, (x_{n_{\mathrm{te}}}^{\mathrm{te}}, y_{n_{\mathrm{te}}}^{\mathrm{te}})\right\} \subseteq \mathcal{X} \times \mathcal{Y}$
- **Dataset Shift**: $P_{tr}(x, y) \neq P_{te}(x, y)$
- Different from **transfer learning** where learning from a variety of previous environments help in learning, inference, and prediction on a new environment.

Assumption 8.1 *We make the simplifying assumption that* $\mathrm{P}_{\mathrm{tr}}(x, y)$ *and* $\mathrm{P}_{\mathrm{te}}(x, y)$ *only differ via* $\mathrm{P}_{\mathrm{tr}}(x, y) = \mathrm{P}(y|x)\mathrm{P}_{\mathrm{tr}}(x)$ *and* $\mathrm{P}_{\mathrm{te}}(x, y) = \mathrm{P}(y|x)\mathrm{P}_{\mathrm{te}}(x)$. *In other words, the conditional probabilities of* $y|x$ *remain* unchanged.

Examples of Dataset Shift

- Diagnose breast cancer:
  - Training: middle-aged women, likely to have breast screening in the preceding three years. (i.e. older woman with low risk of cancer)
  - Sample Selection Bias:
    - Few disease cases: (bias in $P_{tr}(y|x)$)
    - Mostly old women: (bias in $P_{tr}(x)$)
- Gene expression profile studies for tumor diagnosis:
  - Test: recorded under different experimental conditions
    - Covariate shift: ($P_{tr}(x) \neq P_{te}(x)$)

Importance Sampling

- Expected Risk $R[\mathrm{P}, \theta, l(x, y, \theta)] = \mathbf{E}_{(x,y)\sim \mathrm{P}}\left[l(x, y, \theta)\right]$
- Using importance sampling, we know that:

$$R[\mathrm{P}_{\mathrm{te}}, \theta, l(x, y, \theta)] = \mathbf{E}_{(x,y)\sim \mathrm{P}_{\mathrm{te}}}\left[l(x, y, \theta)\right] = \mathbf{E}_{(x,y)\sim \mathrm{P}_{\mathrm{tr}}}\Big[\underbrace{\frac{\mathrm{P}_{\mathrm{te}}(x, y)}{\mathrm{P}_{\mathrm{tr}}(x, y)}}_{:=\beta(x,y)} l(x, y, \theta)\Big]$$

$$= R[\mathrm{P}_{\mathrm{tr}}, \theta, \beta(x, y) l(x, y, \theta)]$$

assuming $\mathrm{support}(P_{te}) \subset \mathrm{support}(P_{tr})$

Problems of Importance Sampling

1. Good estimators of both P_{te} and P_{tr} are needed.
2. We may prefer to estimate $\beta(x, y)$ directly, incorporating regularization and prior knowledge directly.
3. Importance samples weights β which deviate strongly from 1 increase the variance significantly.

Kernel Mean Matching (aka Distribution Matching)

Let $\Phi : \mathcal{X} \to \mathcal{F}$ be a feature map into a feature space $\mathcal{F}$ and denote by $\mu : \mathcal{P} \to \mathcal{F}$ the expectation operator

$$\mu(\mathrm{P}) := \mathbf{E}_{x \sim \mathrm{P}(x)} [\Phi(x)]$$

Define the **Marginal Polytope**: $\mathcal{M}(\Phi) := \{\mu(\mathrm{P}) \text{ where } \mathrm{P} \in \mathcal{P}\}$

(i.e. image of $\mathcal{P}$ under μ)

Theorem 8.2 *The operator μ is a* bijection *between the space of all probability measures and the marginal polytope induced by the feature map $\Phi(x)$ if $\mathcal{F}$ is an RKHS with a universal kernel $k(x, x') = \langle \Phi(x), \Phi(x') \rangle$ in the sense of Steinwart [2002] (bearing in mind that universality is defined for kernels on compact domains $\mathcal{X}$).*

Lemma 8.3 *The following optimization problem in β is convex.*

$$\underset{\beta}{\text{minimize}} \quad \left\| \mu(\mathrm{P}_{\text{te}}) - \mathbf{E}_{x \sim \mathrm{P}_{\text{tr}}(x)} \left[\beta(x) \Phi(x) \right] \right\| \tag{8.13}$$

$$\text{subject to } \beta(x) \geq 0 \text{ and } \mathbf{E}_{x \sim \mathrm{P}_{\text{tr}}(x)} \left[\beta(x) \right] = 1. \tag{8.14}$$

Assume P_{te} *is absolutely continuous with respect to* P_{tr} *(so* $\mathrm{P}_{\text{tr}}(A) = 0$ *implies* $\mathrm{P}_{\text{te}}(A) = 0$*), and that* k *is universal. The solution of (8.13) is then* $\mathrm{P}_{\text{te}}(x) = \beta(x) \mathrm{P}_{\text{tr}}(x)$.

Empirical KMM Optimization

$$\underset{\beta}{\text{minimize}} \quad \left\| \frac{1}{n_{\text{tr}}} \sum_{i=1}^{n_{\text{tr}}} \beta_i \Phi(x_i^{\text{tr}}) - \frac{1}{n_{\text{te}}} \sum_{i=1}^{n_{\text{te}}} \Phi(x_i^{\text{te}}) \right\|^2$$

$$\text{subject to constraints } \beta_i \in [0, B] \text{ and } \left| \tfrac{1}{n_{\text{tr}}} \textstyle\sum_{i=1}^{n_{\text{tr}}} \beta_i - 1 \right| \leq \epsilon$$

- After some manipulation, we can rewrite it as a quadratic program

For $\underset{\beta}{\text{minimize}} \ \frac{1}{2}\beta^\top K \beta - \kappa^\top \beta$

$$\text{subject to } \beta_i \in [0, B] \text{ and } \left| \sum_{i=1}^{n_{\text{tr}}} \beta_i - n_{\text{tr}} \right| \leq n_{\text{tr}} \epsilon.$$

$$\kappa_i := \frac{n_{\text{tr}}}{n_{\text{te}}} \sum_{j=1}^{n_{\text{te}}} k(x_i^{\text{tr}}, x_j^{\text{te}})$$

$$K_{ij} := k(x_i^{\text{tr}}, x_j^{\text{tr}})$$

Example 1: Polynomial Regression

$$y = -x + x^3 + \varepsilon, \quad \varepsilon \sim \mathrm{N}(0, 0.3^2).$$

$$x \sim \mathrm{N}(\mu_0, \tau_0^2), \quad \mathrm{P_{tr}} \sim \mathcal{N}(0.5, 0.5^2) \text{ and } \mathrm{P_{te}} \sim \mathcal{N}(0, 0.3^2)$$

- Sample: 100 training (darker circles) and testing (lighter crosses) points from P_{tr} and P_{te} respectively.
- Fitting Model: Simple linear regression (degree-1 polynomial)
- Solid gray line := OLS on training data (Refence only)
- Competing models:= WOLS with differente weigthing schemes:

  ratio of densities, KMM and information criterion
- KMM:

$$\underset{\theta}{\text{minimize}} \sum_{i=1}^{n_{\mathrm{tr}}} \beta_i (y_i^{\mathrm{tr}} - \langle \Phi(x_i^{\mathrm{tr}}), \theta \rangle)^2 + \lambda \|\theta\|^2 .$$

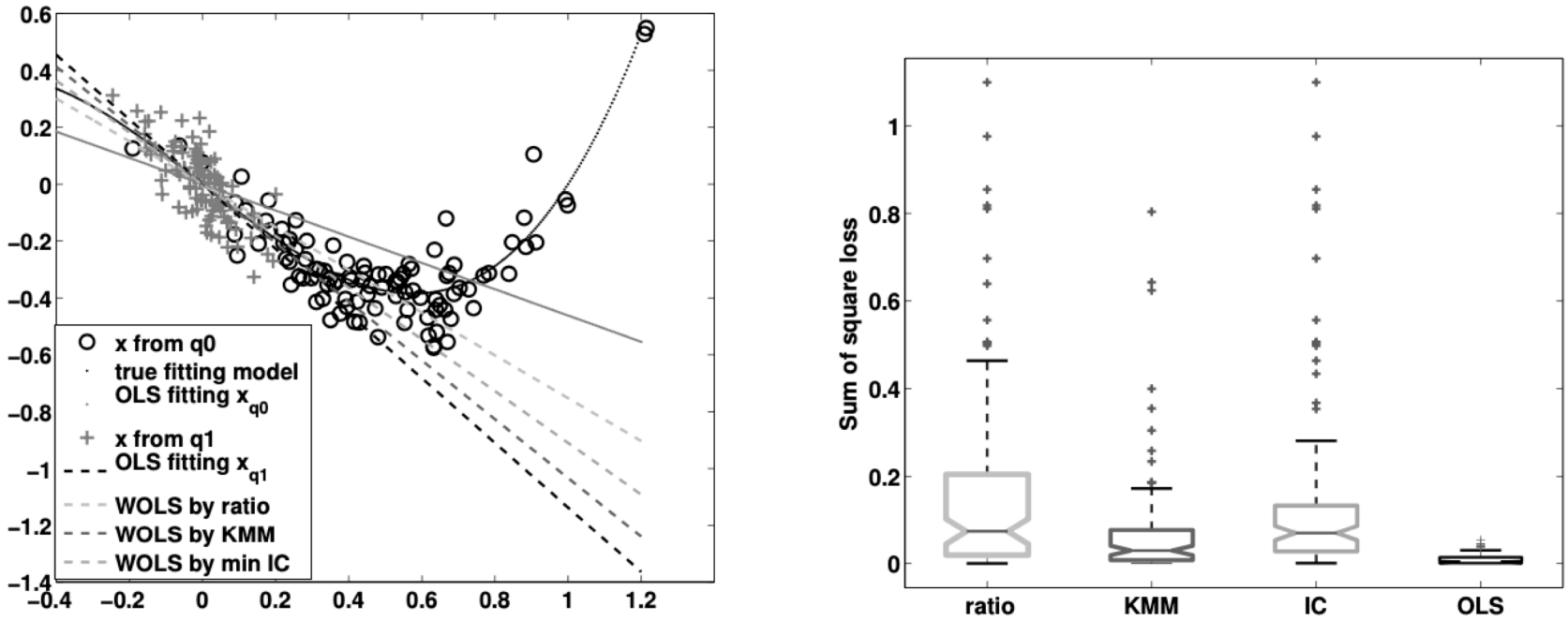


Figure 8.1 *Left:* Polynomial models of degree 1 fit with OLS and WOLS; *Right:* Average performances of three WOLS methods and OLS on this example. Labels are *ratio* for ratio of test to training density; KMM for our approach; *min IC* for the approach of Shimodaira [2000]; and *OLS* for the model trained on the labeled test points.

Breast Cancer Dataset:

- Support Vector Classifier with Slack Re-Scaling (Tsochantaridis et al. [2005])

$$\underset{\theta,\xi}{\text{minimize}} \ \frac{1}{2}\|\theta\|^2 + C\sum_{i=1}^{n_{\text{tr}}} \beta_i \xi_i \tag{8.4a}$$

$$\text{subject to } \left\langle \Phi(x_i^{\text{tr}}, y_i^{\text{tr}}) - \Phi(x_i^{\text{tr}}, y), \theta \right\rangle \geq 1 - \xi_i / \Delta(y_i^{\text{tr}}, y) \tag{8.4b}$$

$$\text{for all } \ y \in \mathcal{Y}, \text{ and } \xi_i \geq 0.$$

Where $\Delta(y, y')$ denote a discrepancy function between y and y'.

Dual:

$$\underset{\alpha}{\text{minimize}} \ \frac{1}{2} \sum_{i,j=1;y,y' \in \mathcal{Y}}^{n_{\text{tr}}} \alpha_{iy}\alpha_{jy'} k(x_i^{\text{tr}}, y, x_j^{\text{tr}}, y') - \sum_{i=1;y\in\mathcal{Y}}^{n_{\text{tr}}} \alpha_{iy} \tag{8.5a}$$

$$\text{subject to } \alpha_{iy} \geq 0 \text{ for all } i, y \text{ and } \sum_{y\in\mathcal{Y}} \alpha_{iy}/\Delta(y_i^{\text{tr}}, y) \leq \beta_i C. \tag{8.5b}$$

$$k(x, y, x', y') := \langle \Phi(x, y), \Phi(x', y') \rangle$$

Breast Cancer Dataset:

- Predictors $x_1, \ldots, x_9$ were normalized to zero mean and unit standard deviation before any other procedure was applied
- Competing cases: unweighted, importance sampling
- Gaussian kernel $\exp(-|x_i - x_j|^2/(2\sigma^2))$
- We fix the kernel size to σ = 5, and vary C over the range C ∈ {0.01, 0.1, 1, 10, 100}.
- Test results always represent an average over 15 splits between training and test set.
- In all cases, an initial split between training and test set is done.

Biased sampling scheme:

- Smaller feature values predominate in the unbiased data
- Test set subsampled according to P(chosen|x ≤ 5) = 0.2 and P(chosen|x > 5) = 0.8
- This subsampling was repeated for each of the features in turn.

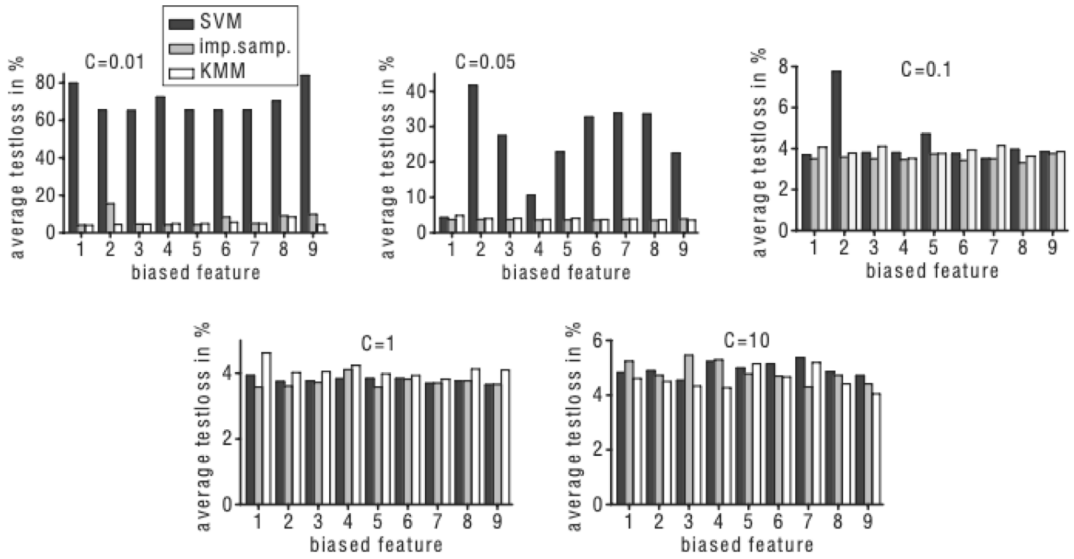


Figure 8.2 Classification performance on UCI breast cancer data. An individual feature bias scheme was used. Test error is reported on the y-axis, and the feature being biased on the x-axis.

Joint sampling sampling scheme:

- Smaller feature values predominate in the unbiased data
- Subsampled Training set: samples less often when they were further from the sample mean $\bar{x}$ over the training data

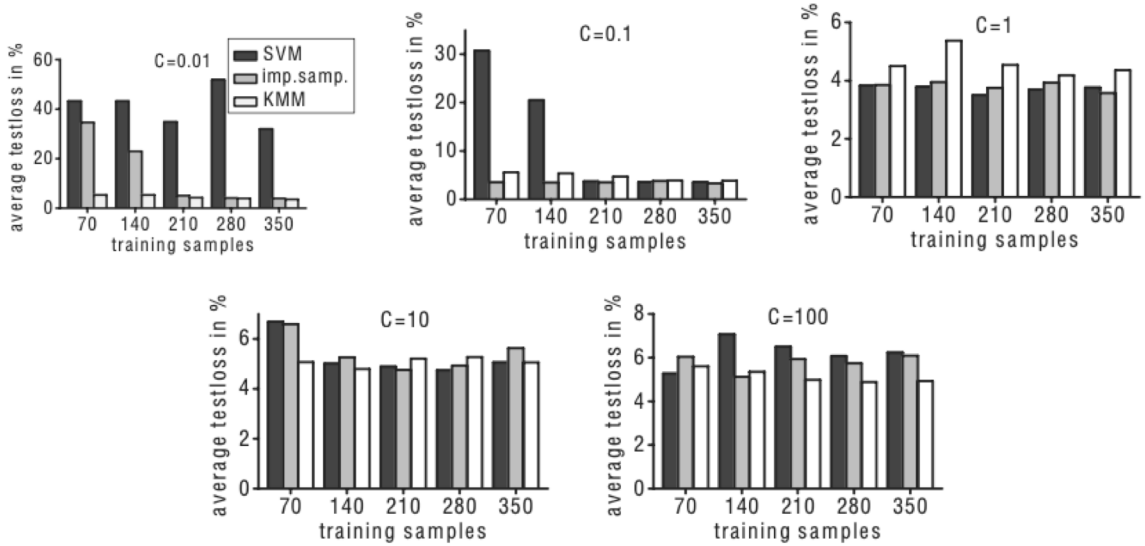


Figure 8.3 Classification performance on UCI breast cancer data. A joint feature bias scheme was used. Test error is reported on the y-axis, and the initial number of training points (prior to biased training point selection) on the x-axis.

Label biased:

- Subsampled Training set: P(chosen|y = 1) = 0.1 and P(chosen|y=-1) = 0.9

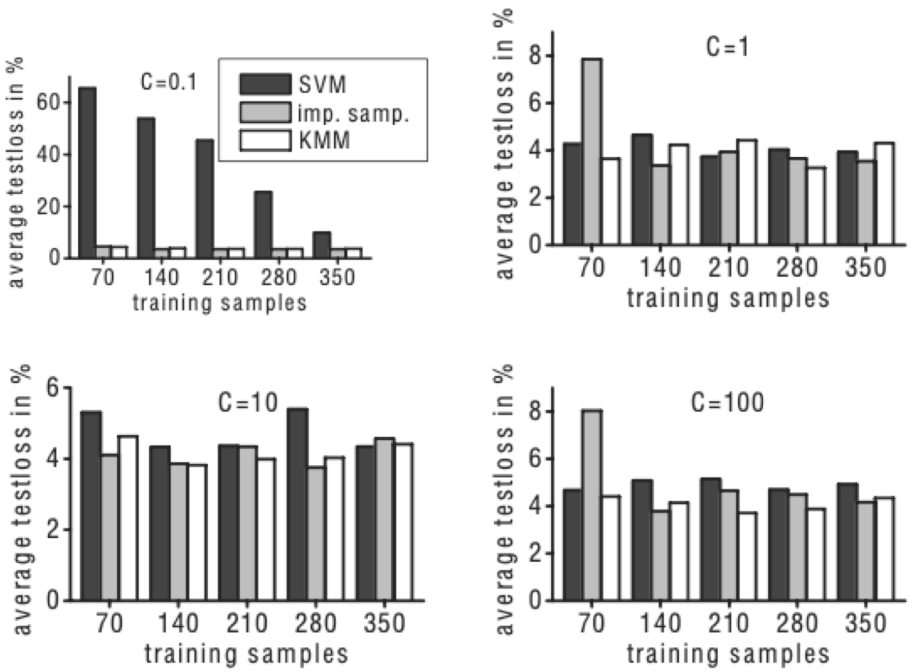


Figure 8.4 Classification performance on UCI breast cancer data. A label bias scheme was used. Test error is reported on the y-axis, and the initial number of training points (prior to biased training point selection) on the x-axis.

- In all three of the above examples, by far the greatest performance advantage for both importance sampling and KMM-based reweighting is for small values of C (and thus, for classifiers which put a high priority on a smooth decision boundary)
- This advantage also holds for bias over the labels, despite this violating our key assumption 8.1.
- We conclude that for the UCI breast cancer data, covariate shift correction (whether by importance sampling or KMM) has the advantage of widening the range of C values for which good performance can be expected (and in particular, greatly enhancing performance at the lowest C levels), at the risk of slightly worsening performance at the optimal C range.

Appendix: Information Criterion (Schimodaira)

- Maximum weighted log-likelihood estimate (MWLE)

Let the information criterion for MWLE be

$$\mathrm{IC}_w := -2L_1(\hat{\theta}_w) + 2\,\mathrm{tr}(J_w H_w^{-1}), \tag{5.1}$$

where

$$L_1(\theta) = \sum_{t=1}^{n} \frac{q_1(x)}{q_0(x)} \log p(y_t|x_t, \theta),$$

$$J_w = -E_0 \left\{ \frac{q_1(x)}{q_0(x)} \frac{\partial \log p(y|x,\theta)}{\partial \theta}\bigg|_{\theta_w^*} \frac{\partial l_w(x,y|\theta)}{\partial \theta'}\bigg|_{\theta_w^*} \right\}.$$

The matrices J_w and H_w may be replaced by their consistent estimates

$$\hat{J}_w = -\frac{1}{n}\sum_{t=1}^{n} \frac{q_1(x_t)}{q_0(x_t)} \frac{\partial \log p(y_t|x_t,\theta)}{\partial \theta}\bigg|_{\hat{\theta}_w} \frac{\partial l_w(x_t,y_t|\theta)}{\partial \theta'}\bigg|_{\hat{\theta}_w},$$

$$\hat{H}_w = \frac{1}{n}\sum_{t=1}^{n} \frac{\partial^2 l_w(x_t,y_t|\theta)}{\partial\theta\partial\theta'}\bigg|_{\hat{\theta}_w}.$$

References

- Quinonero – Dataset Shift in Machine Learning