# STT 873 HW2 (Solution Keys)

This HW is due on Sep 25th.

**Question 1**: Prove the inner product $\langle \cdot, \cdot \rangle$ defined in the RKHS is a well defined inner product by verifying the following properties

(1) Symmetric: $\langle f, g \rangle = \langle g, f \rangle$. *(3 pts)*

(2) Linear: $\langle cf + dg, h \rangle = c\langle f, h \rangle + d\langle g, h \rangle$. *(3 pts)*

(3) $\langle f, f \rangle = 0$ iff $f = 0$. *(4 pts)*

**Solution:** Let

$$f(\cdot) = \sum_{i=1}^{n} \alpha_i k(\cdot, x_i), \quad g(\cdot) = \sum_{j=1}^{m} \beta_j k(\cdot, y_j) \text{ and } h(\cdot) = \sum_{l=1}^{r} \gamma_l k(\cdot, z_l)$$

(1) Note that $k$ is symmetric, i.e, $k(x, y) = k(y, x)$. Then

$$\begin{aligned} \langle f, g \rangle &= \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(x_i, y_j) \\ &= \sum_{j=1}^{m} \sum_{i=1}^{n} \beta_j \alpha_i k(y_j, x_i) \\ &= \langle g, f \rangle \end{aligned}$$

(2) Note that

$$\begin{aligned} \langle f, g \rangle &= \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_i \beta_j k(x_i, y_j) \\ &= \sum_{j=1}^{m} \beta_j f(y_j) \end{aligned}$$

Then

$$\begin{aligned} \langle cf + dg, h \rangle &= \sum_{l=1}^{r} \gamma_l (cf(z_l) + dg(z_l)) \\ &= c \sum_{l=1}^{r} \gamma_l f(z_l) + d \sum_{l=1}^{r} \gamma_l g(z_l) \\ &= c\langle f, h \rangle + d\langle g, h \rangle \end{aligned}$$

(3) "⇒" Note that $\langle k(\cdot, x), f\rangle = \sum_{i=1}^{n} \alpha_i k(x_i, x) = f(x)$. Then

$$\begin{aligned} |f(x)|^2 &= |\langle k(\cdot,x), f\rangle|^2 \\ &\leq |\langle k(\cdot,x), k(\cdot,x)\rangle^{\frac{1}{2}} \langle f, f\rangle^{\frac{1}{2}}|^2 \quad \text{by Cauchy-Schwarz inequality} \\ &= k(x,x)\langle f, f\rangle \\ &= 0 \end{aligned}$$

which implies $f = 0$, since $|f|^2 \geq 0$.
"⇐"

$$\begin{aligned} \langle f, f\rangle &= \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j k(x_i, x_j) \\ &= \sum_{j=1}^{n} \alpha_j f(x_j) \\ &= 0 \quad \text{since } f = 0. \end{aligned}$$

**Question 2** *(10 pts)*

(1) Write a R function to solve the following RKHS regression problem.

$$\hat{f}_\lambda = \underset{f \in RKHS}{\operatorname{argmin}} \sum_{i=1}^{n} (y_i - f(x_i))^2 + \lambda \|f\|_K^2, \tag{1}$$

where $\|\cdot\|_K$ is the norm defined in the RKHS space with Gaussian kernel ($K(x, y') = \exp(-\|x - y'\|^2/0.25)$), Laplacian kernel $K(x, y') = \exp(-\|x - y'\|)$,or Polynomial kernel ($K(x, y') = (< x, y' > +1)^2$). (You can also compare your results with those using the R pacakge 'KERE').

(2) Download data from 'goo.gl/pkBTsy', and test your function with $\lambda = 0.5$ and $0.01$ for Gaussian and Laplacian kernels and $\lambda = 0.5$ and $0.2$ for Polynomial kernel. Plot $X$ vs $\hat{Y}$ on top of the original data points.

**Solution:** By the *Representation Throrem*, the minimizer $\hat{f}_\lambda$ of equation (1) has the form

$$\hat{f}_\lambda(x) = \sum_{i=1}^{n} \alpha_i K(x_i, x)$$

where $K(s, t)$ is the reporducing kernel. See Wahaba (1990) or Gu (2002) for detail.
Define the observation vector $\boldsymbol{y} = (y_1, \ldots, y_n)^T$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_n)^T$ and $\boldsymbol{K}$ as the $n \times n$ matrix where the $i, j$ entry is $\boldsymbol{K}_{ij} = K(x_i, x_j)$. The minimization of (1) then takes the form

$$\|\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\alpha}\|^2 + \lambda \boldsymbol{\alpha}^T \boldsymbol{K} \boldsymbol{\alpha}.$$

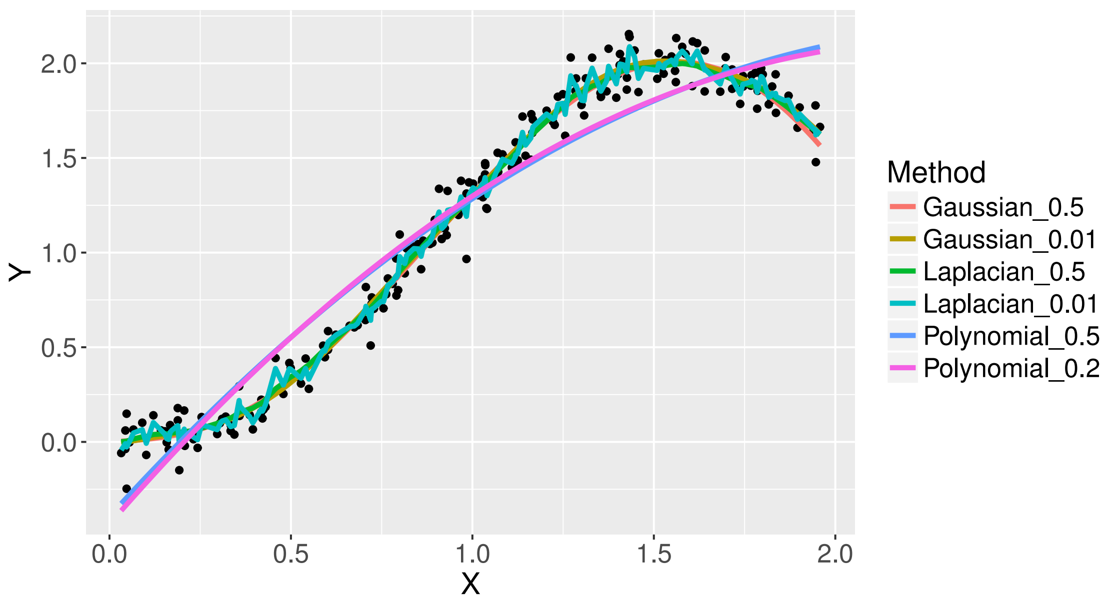


Figure 1: RKHS Regression with specified kernel and $\lambda$.

Taking derviatives with respect to $\boldsymbol{\alpha}$, the minimizer over $\boldsymbol{\alpha}$ is

$$\hat{\boldsymbol{\alpha}} = (\boldsymbol{K} + \lambda \boldsymbol{I})^{-1} \boldsymbol{y}$$

and

$$\hat{f}(x) = \boldsymbol{K}(\boldsymbol{K} + \lambda \boldsymbol{I})^{-1} \boldsymbol{y}.$$

The R code is in the Appendix. The result of the implemented RKHS Regression is plotted in Figure 1. It's clear that Gaussian kernel fits the data very well and there is a small difference between $\lambda = 0.5$ and $\lambda = 0.01$. The Laplacian kernel with $\lambda = 0.5$ also fits the data very well, while with $\lambda = 0.01$ appears to be overfitting. The polynomial kernel does not capture the shape of the data for either $\lambda$ value.