Kernels: Linear models in function space

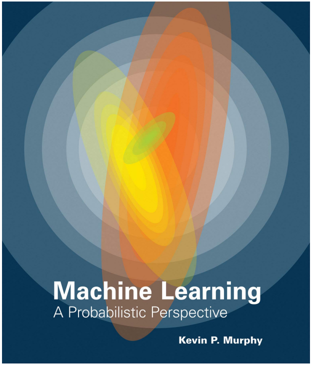
Rodrigo A. Vargas-Hernández

February 7, 2024

References for today's lecture

(31/Jan/2023)

Chapter 13: Sparse linear models and Chapter 14: Kernels



- Kernel ridge regression
- extra material

Regularization Overfitting

- One technique that is often used to control the over-fitting phenomenon in such cases is that of regularization,

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i}^{N}(y_i - \boldsymbol{\theta}^{\mathrm{T}}\phi(\mathbf{x}_i))^2 + \frac{\lambda}{2}\sum_{j} w_j^2 \qquad (1)$$

- λ is known as the **regularization** term.

(HOMEWORK)

- What is the value of the **optimal** parameters $\boldsymbol{\theta}^*$ (Eq. 12)?

(tip): $\sum_j w_j^2 = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}$ and $\frac{\partial \boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}}{\boldsymbol{w}} = 2\boldsymbol{w}$

Notation

- $\mathbf{x} \rightarrow$ single data point, $\mathbf{x} = [x_0, \cdots, x_i, \cdots, x_d]$
- $d \rightarrow$ total number of **features** in $\mathbf{x}$
- $y \rightarrow$ observable of a single point [1]
- $\mathbf{X} \rightarrow$ all data points, $\mathbf{X} = [\mathbf{x}_1^T, \cdots, \mathbf{x}_N^T]$
- $\mathbf{y} \rightarrow$ observables for all data points, $\mathbf{y} = [y_1, \cdots, y_N]$
- $N \rightarrow$ total number of data points
- $\mathcal{D} \rightarrow$ Data set, $\mathcal{D} = [\mathbf{X}, \mathbf{y}]$
- $\boldsymbol{\theta} \rightarrow$ all model's parameters
- $f(\cdot) \rightarrow$ model
- $\mathcal{L}(\cdot) \rightarrow$ loss, error or cost function

[1] Could be a vector, **multiple observables**

Regularization Overfitting

- One technique that is often used to control the over-fitting phenomenon in such cases is that of regularization,

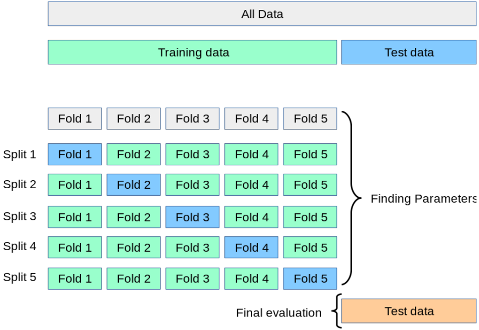
$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\sum_{i}^{N}(y_i - \boldsymbol{\theta}^{\mathrm{T}}\phi(\mathbf{x}_i))^2 + \frac{\lambda}{2}\sum_{j} w_j^2 \tag{2}$$

$$\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right)^{\mathrm{T}}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right) + \frac{\lambda}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} \tag{3}$$

- λ is known as the **regularization** term.
- λ is optimized using Cross-Validation.
- $\boldsymbol{\Phi}(\mathbf{X})$ is the representation of all data points $\mathbf{X}$ in the feature space $\phi(\cdot)$.

(Quick primer) Cross-Validation

- Search algorithm to optimize **hyper-parameters** in ML models.



  - For every fold we search for the best λ.
  - Average all the best λs, $\lambda^* = \frac{1}{K}\sum_j^K \lambda_j$.
  - **What are the cons of CV?**

(code) Cross-Validation

```python
import numpy as np
from sklearn.model_selection import KFold

data = load_data() # data = (X,y)

n_folds = 5 # number of folds
kf = KFold(n_splits=n_folds)

# grid on the possible values of lambda
lambda_grid = np.array([0.,0.001,0.01,0.1,0.5,1.])

l_ = []
# iterate over the k-folds
for train, val in kf.split(X): # index
  X_train, y_train = X[train], y[train]
  X_val, y_Val = X[val], y[val]

  # search algorithm for lambda
  lambda_opt = solve_for_lambda((X_train,y_train), (X_val,y_val),lambda_grid)
  l_.append(lambda_opt)

l_ = np.array(l_)
best_lambda = np.mean(l_)
```

solution of linear regression + regularization

$$\mathcal{L}(\boldsymbol{\theta}) \quad = \quad \frac{1}{2}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right)^{\mathrm{T}}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right) + \frac{\lambda}{2}\boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\theta} \tag{4}$$

- Solve for $\boldsymbol{\theta}$ using $\nabla_{\boldsymbol{\theta}}\mathcal{L}(\boldsymbol{\theta}) = 0$.

Solution: (last week's lecture notes)

$$\boldsymbol{\theta}^{*} \quad = \quad \left(\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X}) + \lambda\mathbb{I}_d\right)^{-1}\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\mathbf{y} \tag{5}$$

- What is $\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})$?
- What are the dimensions of $\boldsymbol{\theta}^{*}$ and $\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})$?

solution of linear regression + regularization

- What is $\mathbf{\Phi}(\mathbf{X})^{\mathrm{T}}\mathbf{\Phi}(\mathbf{X})$?

$$\mathbf{\Phi}(\mathbf{X})^{\mathrm{T}}\mathbf{\Phi}(\mathbf{X}) = \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_0(\mathbf{x}_2) & \cdots & \phi_0(\mathbf{x}_N) \\ \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \cdots & \phi_1(\mathbf{x}_N) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{d-1}(\mathbf{x}_1) & \phi_{d-1}(\mathbf{x}_2) & \cdots & \phi_{d-1}(\mathbf{x}_N) \\ \phi_d(\mathbf{x}_1) & \phi_d(\mathbf{x}_2) & \cdots & \phi_d(\mathbf{x}_N) \end{pmatrix} \begin{pmatrix} \phi_0(\mathbf{x}_1), & \phi_1(\mathbf{x}_1), & \cdots, & \phi_d(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2), & \phi_1(\mathbf{x}_2), & \cdots, & \phi_d(\mathbf{x}_2) \\ & & \cdots & \\ \phi_0(\mathbf{x}_{N-1}), & \phi_1(\mathbf{x}_{N-1}), & \cdots, & \phi_d(\mathbf{x}_{N-1}) \\ \phi_0(\mathbf{x}_N), & \phi_1(\mathbf{x}_N), & \cdots, & \phi_d(\mathbf{x}_N) \end{pmatrix}$$

- $\phi_j(\mathbf{x}_i)$, feature j in $\phi(\cdot)$ for point i

Homework: proof that $\mathbf{\Phi}(\mathbf{X})^{\mathrm{T}}\mathbf{\Phi}(\mathbf{X}) = \sum_i^N \phi(\mathbf{x}_i)\phi(\mathbf{x}_i)^{\mathrm{T}}$
remember, $\phi(\mathbf{x}_i)^{\mathrm{T}} = [\phi_0(\mathbf{x}_i), \phi_1(\mathbf{x}_i), \cdots, \phi_d(\mathbf{x}_i)]$, (vector of $(1, d)$ dimensions)

Kernel space

$$\boldsymbol{\theta}^* = \left(\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X}) + \lambda\mathbb{I}_d\right)^{-1}\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\mathbf{y} \tag{6}$$

Matrix identity (Eq. 167): $(\mathbb{I} + \mathbf{AB})^{-1}\mathbf{A} = \mathbf{A}\left(\mathbb{I} + \mathbf{BA}\right)^{-1}$

$$\boldsymbol{\theta}^{**} = \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}\left(\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} + \lambda\mathbb{I}_N\right)^{-1}\mathbf{y} \tag{7}$$

- What is $\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}$?

$$\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} = \begin{pmatrix} \phi_0(\mathbf{x}_1), & \phi_1(\mathbf{x}_1), & \cdots, & \phi_d(\mathbf{x}_1) \\ \phi_0(\mathbf{x}_2), & \phi_1(\mathbf{x}_2), & \cdots, & \phi_d(\mathbf{x}_2) \\ & & \cdots & \\ \phi_0(\mathbf{x}_{N-1}), & \phi_1(\mathbf{x}_{N-1}), & \cdots, & \phi_d(\mathbf{x}_{N-1}) \\ \phi_0(\mathbf{x}_N), & \phi_1(\mathbf{x}_N), & \cdots, & \phi_d(\mathbf{x}_N) \end{pmatrix} \begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_0(\mathbf{x}_2) & \cdots & \phi_0(\mathbf{x}_N) \\ \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \cdots & \phi_1(\mathbf{x}_N) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{d-1}(\mathbf{x}_1) & \phi_{d-1}(\mathbf{x}_2) & \cdots & \phi_{d-1}(\mathbf{x}_N) \\ \phi_d(\mathbf{x}_1) & \phi_d(\mathbf{x}_2) & \cdots & \phi_d(\mathbf{x}_N) \end{pmatrix}$$

- What are the matrix elements of $\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}$, $[\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{x}_j)]_{ij}$?

Kernel space as linear model

Solution of standard linear regression,

$$
\begin{aligned}
f(\mathbf{x}, \boldsymbol{\theta}^*) &= \sum_i^d \theta_i^* \phi(\mathbf{x})^i = \boldsymbol{\theta}^{*\,\mathrm{T}} \phi(\mathbf{x}) && (8) \\
&= \left[\left(\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} \boldsymbol{\Phi}(\mathbf{X}) + \lambda \mathbb{I}_d \right)^{-1} \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} \mathbf{y} \right]^{\mathrm{T}} \phi(\mathbf{x}) && (9)
\end{aligned}
$$

OTHER solution,

$$
\begin{aligned}
f(\mathbf{x}, \boldsymbol{\theta}^{**}) &= \boldsymbol{\theta}^{**\,\mathrm{T}} \phi(\mathbf{x}) = \phi(\mathbf{x})^{\mathrm{T}} \boldsymbol{\theta}^{**} && (10) \\
&= \underbrace{\phi(\mathbf{x})^{\mathrm{T}} \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}}_{\boldsymbol{\kappa}^{\mathrm{T}}} \underbrace{\left(\boldsymbol{\Phi}(\mathbf{X}) \boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} + \lambda \mathbb{I}_d \right)^{-1} \mathbf{y}}_{\alpha} && (11)
\end{aligned}
$$

Kernel space as linear model

$$f(\mathbf{x},\boldsymbol{\theta}^{**}) = \underbrace{\phi(\mathbf{x})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}}_{\boldsymbol{\kappa}^{\mathrm{T}}}\underbrace{\left(\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} + \lambda\mathbb{I}_d\right)^{-1}\mathbf{y}}_{\alpha} \quad (12)$$

$$= \phi(\mathbf{x})^{\mathrm{T}}\boldsymbol{\theta}^{**} = \boldsymbol{\kappa}^{\mathrm{T}}\alpha = \sum_{i}^{N}\boldsymbol{\kappa}(\mathbf{x},\mathbf{x}_i)\alpha_i \quad (13)$$

$$(14)$$

- What is $\phi(\mathbf{x})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}$ and/or $\boldsymbol{\Phi}(\mathbf{X})\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}}$?

$$\phi(\mathbf{x})^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})^{\mathrm{T}} = \left(\phi_0(\mathbf{x}), \quad \phi_1(\mathbf{x}), \quad \cdots, \quad \phi_d(\mathbf{x})\right)\begin{pmatrix} \phi_0(\mathbf{x}_1) & \phi_0(\mathbf{x}_2) & \cdots & \phi_0(\mathbf{x}_N) \\ \phi_1(\mathbf{x}_1) & \phi_1(\mathbf{x}_2) & \cdots & \phi_1(\mathbf{x}_N) \\ \vdots & \vdots & \vdots & \vdots \\ \phi_{d-1}(\mathbf{x}_1) & \phi_{d-1}(\mathbf{x}_2) & \cdots & \phi_{d-1}(\mathbf{x}_N) \\ \phi_d(\mathbf{x}_1) & \phi_d(\mathbf{x}_2) & \cdots & \phi_d(\mathbf{x}_N) \end{pmatrix}$$

$$= \left(\phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{x}_0), \quad \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{x}_1), \quad \cdots, \quad \phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{x}_N)\right)$$

- Do we need to compute $\phi(\mathbf{x}_i)^{\mathrm{T}}\phi(\mathbf{x}_j)$?

Kernel function

- $\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^{\mathrm{T}} \phi(\mathbf{x}_j)$
  - $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ is the dot or inner product between functions,

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle_\nu \tag{15}$$

- (From Wikipedia): On the other hand, an explicit representation for φ is not necessary, as long as ν is an inner product space.

- For $\phi(\cdot)$ as an infinite polynomial, $\kappa(\mathbf{x}_i, \mathbf{x}_j)$ is the **Squared Exponential Kernel**,

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2 \exp^{-\frac{(\mathbf{x}_i - \mathbf{x}_j)^2}{2\ell^2}} \tag{16}$$

- How are we optimizing σ and ℓ ?
- What are the **free-parameters** of this class of models?

Kernel functions

The Kernel Cookbook by David Duvenaud

Sklearn tutorial

Faster optimization of kernel ridge regression

Lasso regression

MUST STUDY THIS ON YOUR OWN!

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta}) &= \frac{1}{2}\sum_{i}^{N}(y_i - \boldsymbol{\theta}^{\mathrm{T}}\phi(\mathbf{x}_i))^2 + \underbrace{\frac{\lambda}{2}\sum_{j}|w_j|}_{\text{absolute value}} \\ &= \frac{1}{2}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right)^{\mathrm{T}}\left(\mathbf{y} - \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{\Phi}(\mathbf{X})\right) + \underbrace{\frac{\lambda}{2}\|\boldsymbol{w}\|_1}_{\text{regularization}}\end{aligned}$$

Useful links:

- Wikipedia
- original paper
- Sklearn tutorial

- What is Lasso used for?