Statistical Learning, Fall 2024-5

Class notes 11

Model evaluation and selection

Last week:

- Basic definitions: selection, evaluation,
- Sampling-based: Training-validation-test, K-fold cross-validation (CV), n-fold or leave-one-out CV (LOOCV)

LOOCV and the leaving out lemma

The Leaving Out Lemma requires two conditions under the squared loss, iid error assumption:

1. Linear model: $\hat{\mathbb{Y}} = S(\mathbb{X})\mathbb{Y}$ in training.
2. For any $1 \leq i_0 \leq n$, define a pseudo training dataset $\mathbb{X}, \tilde{\mathbb{Y}}$ with the same $\mathbb{X}$ as our training data, and $\tilde{y}_j = y_j$ for $j \neq i_0$ and:
$$\tilde{y}_{i_0} = \hat{y}_{i_0}^{(-i_0)},$$
where the superscript $(-i_0)$ indicates the model built on $n-1$ observations, leaving out i_0. Then we require:
$$\hat{\tilde{y}}_{i_0} = (S\tilde{\mathbf{y}})_{i_0} = \hat{y}_{i_0}^{(-i_0)}.$$

Under these conditions we can easily prove that:

$$(y_{i_0} - \hat{y}_{i_0}^{(-i_0)}) = \frac{(y_{i_0} - \hat{y}_{i_0})}{1 - S_{i_0 i_0}},$$

and therefore:

$$LOOCV = \sum_{i=1}^{n} \frac{(y_i - \hat{y}_i)^2}{(1 - S_{ii})^2},$$

and we can calculate LOOCV by only fitting the model on the training data once.

Proofs:

- **OLS complies with condition 2:** Define in the obvious way $\hat{\beta}$ as the full OLS solution and $\hat{\beta}^{-(i_0)}$ as the OLS solution with observation i_0 held out:

$$\hat{\beta}^{-(i_0)} = \arg\min_{\beta} \sum_{i=1, i\neq i_0}^{n} (Y_i - x_i^t\beta)^2.$$

Now define $\tilde{\mathbb{Y}}$ as in condition 2:

$$\tilde{Y}_i = \begin{cases} Y_i & \text{if } i \neq i_0 \\ x_{i_0}^t \hat{\beta}^{-(i_0)} & \text{if } i = i_0 \end{cases}$$

Now fit OLS to $(\mathbb{X}, \tilde{\mathbb{Y}})$:

$$\tilde{\hat{\beta}} = \arg\min_{\beta} \sum_{i=1, i\neq i_0}^{n} (Y_i - x_i^t\beta)^2 + (x_{i_0}^t(\hat{\beta}^{-(i_0)} - \beta))^2,$$

the first term is minimized by $\beta = \hat{\beta}^{-(i_0)}$ and the second is 0 when $\beta = \hat{\beta}^{-(i_0)}$. Therefore $\tilde{\hat{\beta}} = \hat{\beta}^{-(i_0)}$.

- **Proof of the Lemma under the conditions:**

$\hat{Y}_{i_0} = (S\mathbb{Y})_{i_0} = \sum_{j=1}^{n} S_{i_0 j} Y_j$, From property 2: $\hat{Y}^{(-i_0)} i_0 = (S\tilde{\mathbb{Y}})_{i_0} = \sum_{j=1, j\neq i_0}^{n} S_{i_0 j} Y_j + S_{i_0 i_0} \hat{Y}^{(-i_0)} i_0$.

Therefore:

$$Y_{i_0} - \hat{Y}_{i_0}^{(i_0)} = Y_{i_0} - \hat{Y}_{i_0} + S_{i_0 i_0}(Y_{i_0} - \hat{Y}_{i_0}^{(i_0)}).$$

Changing sides we get:

$$Y_{i_0} - \hat{Y}_{i_0} = (1 - S_{i_0 i_0})(Y_{i_0} - \hat{Y}_{i_0}^{(i_0)}).$$

Ridge regression also complies with the two conditions. K-NN complies with the first but not the second, while we stated that Lasso does not comply with the result, but left it as a challenge to figure out which of the conditions do not hold.

Optimism and degrees of freedom

Recall for Fixed-X, we have the training squared loss: $\frac{1}{n} RSS = \frac{1}{n} \|\mathbb{Y} - \hat{\mathbb{Y}}\|^2$, and the prediction loss for $\mathbb{Y}^{new}$ an iid copy of $\mathbb{Y}$ at same $\mathbb{X}$: $\mathbb{E}_{\mathbb{Y}, \mathbb{Y}^{new}} \frac{1}{n} \|\mathbb{Y}^{new} - \hat{\mathbb{Y}}\|^2$ (where $\mathbb{Y}$ plays a role through defining $\hat{\mathbb{Y}}$).

It is natural to define the *optimism* of a model building approach (a mapping $\mathbb{Y} \to \hat{\mathbb{Y}}$) as the difference in expectation between the two measures:

$$op = \mathbb{E}_{\mathbb{Y}, \mathbb{Y}^{new}} \frac{1}{n} \|\mathbb{Y}^{new} - \hat{\mathbb{Y}}\|^2 - \frac{1}{n} \|\mathbb{Y} - \hat{\mathbb{Y}}\|^2.$$

The beautiful and fundamental result is that under very general assumptions we have:

$$op = \frac{2}{n} \sum_{i=1}^{n} Cov(y_i, \hat{y}_i).$$

Furthermore, we can actually calculate or estimate this quantity for many models of interest. If we can calculate op or an estimate $\hat{op}$, then we can use it to obtain an unbiased estimate of the prediction error as :

$$\frac{1}{n}RSS + op.$$

Proof of optimism formula:

$$\begin{aligned}
\mathbb{E}\|\mathbb{Y}-\hat{\mathbb{Y}}\|^2 &= \mathbb{E}\|\mathbb{Y}-\mathbb{E}\mathbb{Y}+\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}+\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2 = \mathbb{E}\|\mathbb{Y}-\mathbb{E}\mathbb{Y}\|^2 + \|\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}\|^2 + \mathbb{E}\|\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2 + \\
&+ \underbrace{2\mathbb{E}(\mathbb{Y}-\mathbb{E}\mathbb{Y})^T(\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}})}_{A} + \underbrace{2\mathbb{E}(\mathbb{Y}-\mathbb{E}\mathbb{Y})^T(E\hat{\mathbb{Y}}-\hat{\mathbb{Y}})}_{B} + \underbrace{2(\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}})^T\mathbb{E}(\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}})}_{C} = \\
&= \underbrace{\|\mathbb{Y}-\mathbb{E}\mathbb{Y}\|^2}_{\text{Irreducible}} + \underbrace{\|\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}\|^2}_{\text{Bias}^2} + \underbrace{\mathbb{E}\|\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2}_{\text{Variance}} - 2\underbrace{\mathbb{E}(\mathbb{Y}-\mathbb{E}\mathbb{Y})^T(\hat{\mathbb{Y}}-\mathbb{E}\hat{\mathbb{Y}})}_{\text{Covariance}}
\end{aligned}$$

It is easy to see that $A = C = 0$ from arguments we have seen previously. For the prediction error (remember that $\mathbb{Y}^{new}, \mathbb{Y}$ are identically distributed and in particular $\mathbb{E}\mathbb{Y}^{new} = \mathbb{E}\mathbb{Y}$):

$$\begin{aligned}
\mathbb{E}\|\mathbb{Y}^{new}-\hat{\mathbb{Y}}\|^2 &= \mathbb{E}\|\mathbb{Y}^{new}-\mathbb{E}\mathbb{Y}+\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}+\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2 = \mathbb{E}\|\mathbb{Y}-\mathbb{E}\mathbb{Y}\|^2 + \|\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}\|^2 + \mathbb{E}\|\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2 + \\
&+ \underbrace{2\mathbb{E}(\mathbb{Y}^{new}-\mathbb{E}\mathbb{Y})^T(\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}})}_{A} + \underbrace{2\mathbb{E}(\mathbb{Y}^{new}-\mathbb{E}\mathbb{Y})^T(E\hat{\mathbb{Y}}-\hat{\mathbb{Y}})}_{B} + \underbrace{2(\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}})^T\mathbb{E}(\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}})}_{C} = \\
&= \underbrace{\|\mathbb{Y}-\mathbb{E}\mathbb{Y}\|^2}_{\text{Irreducible}} + \underbrace{\|\mathbb{E}\mathbb{Y}-\mathbb{E}\hat{\mathbb{Y}}\|^2}_{\text{Bias}^2} + \underbrace{\mathbb{E}\|\mathbb{E}\hat{\mathbb{Y}}-\hat{\mathbb{Y}}\|^2}_{\text{Variance}}
\end{aligned}$$

And it follows that:

$$op = \frac{1}{n}\left(\mathbb{E}\|\mathbb{Y}^{new}-\hat{\mathbb{Y}}\|^2 - \mathbb{E}\|\mathbb{Y}-\hat{\mathbb{Y}}\|^2\right) = \frac{2}{n}\sum_{i=1}^{n} Cov(y_i, \hat{y}_i) = \frac{2}{n}tr(Cov(\mathbb{Y}, \hat{\mathbb{Y}})).$$

The simple setting where we can use this result is when:

- We have iid error model $y = f(X) + \epsilon\ ,\ \epsilon \sim (0, \sigma^2)$
- We have a linear model (in the generalized sense): $\hat{\mathbb{Y}} = S(\mathbb{X})\mathbb{Y}$.

(notice no linearity or normality assumptions). In these cases we can write:

$$op = \frac{2}{n}tr(Cov(\mathbb{Y}, S\mathbb{Y})) = \frac{2}{n}tr(SCov(\mathbb{Y}, \mathbb{Y})) = \frac{2\sigma^2}{n}tr(S\mathbb{I}) = \frac{2\sigma^2 tr(S)}{n}.$$

For least squares: $S = H = \mathbb{X}\left(\mathbb{X}^T\mathbb{X}\right)^{-1}\mathbb{X}^T$, so:

$$op = \frac{2\sigma^2}{n}tr(H) = \frac{2p}{n}\sigma^2,$$

in other words, an unbiased estimate of Fixed-X prediction error is:

$$\frac{RSS(\hat{\beta})}{n} + \frac{2p\sigma^2}{n}.$$

For ridge regression: $S_\lambda = \mathbb{X}\left(\mathbb{X}^T\mathbb{X} + \lambda\mathbb{I}\right)^{-1}\mathbb{X}^T$ for $\lambda > 0$, so:

$$op = \frac{2\sigma^2}{n} tr(S_\lambda) < \frac{2p}{n}\sigma^2,$$

and we see the reduced optimism from adding regularization.

We can also apply this result to k-NN which has the required form for Fixed-X, the result? in HW4...

Interesting extensions we may discuss as time permits:

- Dealing with unknown σ^2 (e.g. using unbiased estimates assuming linear model)
- Extending beyond squared loss to likelihood-loss (as in logistic regression): AIC
- Extending to cases where *op* cannot be calculated but can be estimated in unbiased manner: Stein's Lemma

Kernel methods

The general paradigm we have discussed, given modeling problem with $x \in \mathbb{R}^p$ low dimensional:

- Embed $x \to h(x) \in \mathbb{R}^q$ with $q >> p$.
- Fit a (possible linear model) in the high dimension $\hat{f}(x) = \sum_{j=1}^{q} h_j(x)\hat{\beta}_j$.
- Challenges:
  - Computational: how to fit in high dimension
  - Statistical: how to regularize in high dimension

Examples:

- Boosting:
  - Model space: all trees of given size
  - Computational trick: coordinate descent via gradient boosting
  - Regularization: sort of lasso (not discussed in class)
- DNN:
  - Model space: Not a linear model but linear combination of non-linear transformation of linear combinations...
  - Computational tricks: (stochastic) gradient descent,
  - Regularization: sort of ridge (gradient descent $\approx$ ridge, similarly dropout

Now we will discuss perhaps the primary example of this thinking, which was hugely important in ML in the past, lost some of its glamour: Kernel methods including (but not limited to) kernel SVM. We can think of the basic idea the same way, except now $x \to h(x)$ where $h_1, ...h_q$ (possibly $q = \infty$) is a basis of a Reproducing kernel Hilbert functional space (RKHS) $\mathcal{H}_K$. The space is defined indirectly through the kernel function

$$K(\cdot,\cdot) : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R} \text{ such that: } K(x,y) = \langle h(x), h(y)\rangle = \sum_{j=1}^{q} h_j(x)h_j(y).$$

We also naturally define for a function in $\mathcal{H}_K$, $f = \sum_j \beta_j h_j$, we naturally define $\|f\|^2_{\mathcal{H}_K} = \sum_j \beta_j^2$.

Kernel examples:

1. Linear Kernel ($q = p$): $K(x,y) = \langle x, y\rangle$. Here $\mathcal{H}_K$ is simply linear functions.
2. Polynomial kernel: $K_d(x,y) = (1 + x^t y)^d$. Here $q = \binom{p+d}{p}$ all polynomials in x_j, y_j up to degree d.
3. RBF (Gaussian) kernel: $K_\sigma(x,y) = \exp\left(-\|x-y\|^2/(2\sigma^2)\right)$. Here $q = \infty$ and we usually don't think about $h_1, ...$ explicitly, only about the kernel as measuring distance:
   - When σ is small, the kernel $K(x,\cdot)$ is very tight around x
   - When σ is big, the kernel $K(x,\cdot)$ becomes very spread and $K(x,y)$ remains big for $\|x-y\|$ big

   Since $q = \infty$ the function space $\mathcal{H}_K$ contains all nicely behaved functions regardless of σ, however we will see that the different nature of the kernel will play a role in model building (i.e. selecting among the functions in $\mathcal{H}_K$) through regularization.

Kernel machines

The Hilbert space comes with a norm attached and therefore a natural regularization term that controls that norm. Given a loss function our problem is:

$$\hat{f}_\lambda = \arg\min_{f\in\mathcal{H}_K} \sum_{i=1}^{n} L(y_i, f(x_i)) + \lambda\|f\|^2_{\mathcal{H}_K}.$$

We see here that the regularization term is where the specific kernel plays an important role: how functions in $\mathcal{H}_K$ are prioritized for fitting.

The most important result in this area is *the Representer theorem* (Kimmeldorf and Wahba 1970):
The optimal solution to the kernel regression problem above has the form:

$$\hat{f}_\lambda = \sum_{i=1}^{n} \alpha_i K(x_i,\cdot)\ ,\ \ \|\hat{f}_\lambda\|^2_{\mathcal{H}_K} = \alpha^T K\alpha, \text{ where: } K_{ij} = K(x_i,x_j).$$

Thus we get that we can solve the problem in the n dimensional basis of the columns of K:

$$\hat{f}_\lambda = \arg\min_{\alpha} \sum_{i=1}^{n} L(y_i, \sum_{j=1}^{n} \alpha_j K(x_i,x_j)) + \lambda\alpha^T K\alpha.$$

For squared loss this *Kernal linear regression* problem can be nicely written:

$$\hat{f}_\lambda = \sum \hat{\alpha}_i K(x_i, \cdot) \text{ where: } \hat{\alpha} = \arg\min_\alpha \|\mathbb{Y} - K\alpha\|^2 + \lambda \alpha^T K \alpha,$$

a "generalized ridge regression" problem, with an algebraic solution:

$$\hat{\alpha} = (K + \lambda I_n)^{-1} \mathbb{Y}.$$

Now we can interpret what some of our kernels do in this context:

- Linear kernel: $K = \mathbb{X}\mathbb{X}^T$ and therefore $\hat{\alpha} = (\mathbb{X}\mathbb{X}^T + \lambda I_n)^{-1}\mathbb{Y}$. In this case we can easily show:

  $$K\hat{\alpha} = \mathbb{X}\mathbb{X}^T(\mathbb{X}\mathbb{X}^T + \lambda I_n)^{-1}\mathbb{Y} = \mathbb{X}(\mathbb{X}^T\mathbb{X} + \lambda I_p)^{-1}\mathbb{X}^T\mathbb{Y} = \mathbb{X}\hat{\beta}_\lambda,$$

  the solution is the same as regular ridge regression!

- RBF Kernel with small σ:

  $$K(x_i, x_j) = \exp\left(-\|x_i - x_j\|^2/(2\sigma^2)\right) \approx 0 \text{ when } x_i \neq x_j.$$

  Therefore the kernel regression problem is very much like penalized k-NN:

  $$\|\mathbb{Y} - K\alpha\|^2 + \lambda\alpha^T K\alpha \approx \|\mathbb{Y} - \alpha\|^2 + \lambda\alpha^T\alpha.$$

The most important kernel machine was the one using the hinge loss (kernel SVM):

$$L(y, \hat{y}) = (1 - y\hat{y})_+,$$

and recall that we discussed how the sparsity of the solution $\hat{\alpha}$ helps in computing and finding solution.

For regression, the ML crowd who like loss functions that zero many $\hat{\alpha}$ came up with the ϵ-support vector regression loss, which is absolute loss with a *dontcare* region in the middle:

$$L(y, \hat{y}) = (|y - \hat{y}| - \epsilon)_+,$$

Now we can also describe kernel methods in the high dimensional modeling framework:

- Model space: all functions in the RKHS
- Computational tricks: representer theorem, giving a problem of dimension n; Sparsity of solution $\hat{\alpha}$ ("support vectors") when selecting appropriate loss functions, like hinge loss of SVM for classification or ϵ-SVR for regression
- Regularization: RKHS norm, sort of ridge